



B I MOYLE ASSOCIATES, INC.

5788 LINCOLN DRIVE • MINNEAPOLIS, MN 55436 • 952-933-2885

B I M - E P I C

DISK / TAPE MANAGEMENT SYSTEM

USER REFERENCE

Release 6.3

(COPYRIGHT © 2001, B I MOYLE ASSOCIATES, INC.)

BIM-EPIC is a proprietary product of B I Moyle Associates, Inc. It cannot be reproduced, changed, copied, or stored in any form (including, but not limited to, copies on magnetic media) without the express prior written permission of B I Moyle Associates, Inc.

This documentation applies to
Release 6.3 of the program product
BIM-EPIC.

Original Printing 01/08/2001
Last Revised 10/15/2001

Contents

Trademark Information.....	x
Preface	xi
Related Publications	xiv
Chapter 1 Introduction.....	1-1
BIM-EPIC Fundamentals	1-2
Configuration Options	1-2
Controlled Datasets.....	1-3
Start Track 1 Datasets.....	1-4
Controlling Different Types of Datasets	1-5
DSN Catalog Entries.....	1-6
DSN Catalog Management	1-7
DSN Catalog Maintenance	1-8
BIM-EPIC Dataset Definitions (EDDs)	1-9
Introduction	1-9
EDD Tape Information	1-12
EDD Disk Information.....	1-14
Version Information	1-16
Introduction	1-16
Automatic Cataloging	1-19
Retention and Scratching	1-21
Rules for Dataset Retention	1-22
Examples	1-23
Retention for Start Track 1 Datasets.....	1-24
Overriding Standard Retention	1-25
Automatic Generation Purge.....	1-26
The Recorder File	1-27
Additional BIM-EPIC Features and Functions	1-28
LUB Allocation	1-28
Managing Tape Devices	1-29
Managing Disk Devices.....	1-31
Starting and Stopping Partitions	1-32
Creating Reports	1-33
The BIM-EPIC Environment	1-34
When is BIM-EPIC On?	1-34
What Happens When BIM-EPIC Scratches or Deletes a Dataset?	1-35
BIM-EPIC JCL	1-36
Physical Input/Output Control System and Pre-Open Checking.....	1-37
What happens if the job cancels?.....	1-38
BIM-EPIC Job Control	1-39
Introduction	1-39
Tape Processing.....	1-40
Disk Processing	1-42
Quick Disk Allocation (QDAS)	1-42
Chapter 2 BIM-EPIC Online Manager.....	2-1
Introduction to Manual Cataloging	2-2
Introduction to Online Manager	2-3
Overview.....	2-3

Online Catalog Management Commands.....	2-5
Using the Online Manager.....	2-7
Configuring Online Security	2-7
Maintaining User Profiles	2-12
Customizing the EDD Display	2-13
Customizing the Version Display.....	2-14
Customizing the Confirmation Options.....	2-15
Using Online Manager Help	2-16
Managing BIM-EPIC Dataset Definitions (EDDs)	2-18
Displaying BIM-EPIC Dataset Definitions	2-19
Managing EDDs.....	2-21
Displaying a Dataset Definition	2-22
Creating a Dataset Definition.....	2-23
Updating a Dataset Definition.....	2-25
Locking a Dataset Definition	2-26
Releasing a Locked Dataset Definition	2-27
Unlocking a Dataset Definition.....	2-28
Deleting a Dataset Definition.....	2-29
Managing Dataset Versions.....	2-30
Displaying Version Information.....	2-30
Selecting a Version from a List.....	2-32
Displaying Version Attributes.....	2-33
Adding Version Information.....	2-34
Locking a Version.....	2-35
Releasing a Locked Version	2-36
Unlocking a Version	2-37
Closing a Version.....	2-38
Removing Version Information from the DSN Catalog.....	2-39
Managing Tapes	2-40
How BIM-EPIC Selects Tapes for Mounting	2-41
Displaying Tape Volume Information	2-42
Maintaining Tape Volume Information	2-43
Purging a Tape Volume	2-44
Recording Tape Cleaning	2-45
Retention and Scratching.....	2-46
Understanding Retention Concepts.....	2-46
Setting Dataset Definition Retention.....	2-47
Modifying Dataset Definition Retention.....	2-48
Setting or Modifying Dataset Version Retention	2-49
Cycling a Dataset Version.....	2-50
Scratching a Dataset Version	2-52
Chapter 3 DSN Catalog Management Utility	3-1
Command Summary	3-3
BIM-EPIC System Requirements.....	3-4
BIM-EPIC Requirements.....	3-4
CA-EPIC for CMS Requirements	3-5
TSIDMNT Commands	3-6
Command Syntax	3-6
PASSWORD.....	3-7
ADD.....	3-8
CAPACITY	3-11

CATALOG	3-12
CLEAN	3-18
CLOSE.....	3-19
CYCLE	3-20
DELETE	3-22
DEQ	3-23
DISPLAY	3-24
ENQ	3-27
LOCK	3-28
MCATALOG.....	3-29
MOVE	3-30
OFFLINE (VSE Only).....	3-31
ONLINE (VSE Only)	3-32
PURGE.....	3-33
QUIESCE (VSE Only)	3-34
RELEASE.....	3-35
RENAME	3-36
RETAIN	3-37
SCRATCH.....	3-39
TRANSFER.....	3-41
UNLOCK.....	3-42
UPDATE	3-43
 Chapter 4 Accessing the DSN Catalog	4-1
Displaying DSN Catalog Entries (TSIDUTL DSPLY)	4-2
Accessing DSN Catalog Records (TSIDAC2)	4-5
TSIDAC2 Calling Procedure	4-6
TSIDAC2 Search Arguments and Calls.....	4-8
ADDRESS EPIC.....	4-11
 Chapter 5 BIM-EPIC TLBL and DLBL Statements	5-1
TLBL Statements	5-2
TLBL Options.....	5-6
DLBL Statements.....	5-8
DLBL Options	5-10
 Chapter 6 BIM-EPIC DD Statements	6-1
Specifying the DD Statement	6-2
Introduction	6-2
DD Statement Syntax.....	6-3
Special DD Names.....	6-5
DD Statements for Tape and Disk Datasets	6-7
Introduction to the Parameters	6-7
What to do for Different Types of Datasets	6-9
AMP Parameter	6-12
DCB Parameter.....	6-13
DISP Parameter	6-15
Normal Termination Subparameters.....	6-17
Deleting Datasets Using DDGEN and the DISP Parameter.....	6-18
Abnormal Termination Subparameters	6-21
DSN Parameter	6-22
FREE Parameter	6-24

LABEL Parameter	6-25
OPTION Parameter	6-26
SPACE Parameter	6-30
UNIT Parameter.....	6-33
VOLUME Parameter	6-35
DD Statements For Printer and Punch Datasets	6-38
DD Statements for In-Stream Datasets	6-39
DD Statements for Dummy Datasets	6-41

Chapter 7 Running Jobs With BIM-EPIC..... 7-1

Introduction	7-2
Unsupported Datasets	7-2
JCL or DSN Catalog?	7-3
Input Dataset Selection	7-4
Defining Your Datasets With TLBL, DLBL, and DD Statements	7-5
Rules For Dataset Name	7-6
LUB Allocation and the ASSGN Statement	7-7
Disk Processing.....	7-8
Controlled Datasets.....	7-8
Start Track 1 Datasets	7-9
Reusing the Same Space	7-10
Processing Tape Datasets	7-11
Processing Internal Tape Labels	7-11
Rewind Options and MTC Statements	7-12
Multi-Volume Processing	7-13
Multi-Dataset Processing	7-14
Multi-Dataset Processing Examples.....	7-18
Types of Datasets	7-24
Sequential Datasets	7-24
Direct Access Datasets.....	7-25
ISAM Datasets	7-26
VSAM and VSAM-Managed SAM Datasets.....	7-28
Work Datasets.....	7-29
CPU and Partition Independence	7-30
Common Work Datasets (CWDS)	7-31
TYPEFLE=WORK DTFs.....	7-32
Compiler Work Datasets.....	7-33
Uncontrolled Datasets.....	7-34
Physical IOCS Processing.....	7-35
Sorting with BIM-EPIC.....	7-37
Introduction.....	7-37
BIM-EPIC Sort Interface (TSIDSRT)	7-38
Sort Specifics	7-39
SORTWK Datasets	7-41
IDCAMS BACKUP and RESTORE	7-43
IDCAMS REPRO	7-44
IDCAMS IMPORT/EXPORT.....	7-45
LIBR BACKUP/RESTORE.....	7-46
DITTO.....	7-47
VSE/POWER ACCOUNT FILE.....	7-48
VSE/POWER PDISPLAY, POFFLOAD and Printer/Punch Tape Spooling	7-49
SQL/VSE Tape Output Functions	7-51

VSE/ICCF and Interactive Partition GETVIS Requirements with BIM-EPIC	7-52
Processing with COBOL II	7-53
Using BIM-EPIC's JCL Features	7-54
Restarting Programs from a Checkpoint	7-54
Choosing the Device Type.....	7-56
System Datasets and VSE Device Independence.....	7-59
Dummy Datasets and ASSGN IGNORE	7-60
Concatenating Datasets.....	7-61
Processing Multiple Datasets with the Same DTF Name (Dataset Drop).....	7-63
Running Test Jobs With Production Datasets.....	7-64
Chapter 8 BIM-EPIC Utilities	8-1
Pre-Open Processing (TSIDASS).....	8-2
Using the BIM-EPIC Sort Interface (TSIDSRT).....	8-6
Deleting Datasets (TSIDDEL).....	8-7
Copying and Merging Datasets (TSIDDTD)	8-9
Printing Datasets (TSIDDMPP).....	8-16
Accessing the VSE Label Area (TSIDGLB and TSIDPLB).....	8-21
Chapter 9 Job Management Control Language	9-1
JMCL Statements	9-2
Introduction	9-2
Using JOB, EXEC, and OPTION Statements.....	9-3
Using Comment Statements	9-4
JMCL Parameters.....	9-5
JCL Syntax Checking.....	9-5
Condition Checking and Branching	9-7
Conditions and IF	9-9
Setting Values	9-11
Symbolic Replacements.....	9-12
RESET.....	9-13
JMCL Example	9-14
Using the JMCL Interface (TSIDJC2)	9-15
Introduction	9-15
Retrieving JMCL Data (GET REQUEST).....	9-16
Updating JMCL Data (PUT REQUEST).....	9-17
TSIDJC2 Return Codes	9-19
Executing CP Commands.....	9-20
Chapter 10 Job Scheduling.....	10-1
Event Posting	10-2
Introduction	10-2
Scheduling Functions.....	10-3
Posting an Event and Releasing Matching Jobs (TSIDREL and TSIDCREL)	10-4
Posting an Event Only (TSIDLOC)	10-5
Checking for a Posted Event (TSIDCHK and TSIDCCHK)	10-6
Deleting a Posted Event (TSIDUNL)	10-7
Executing in Batch Jobs.....	10-8
Executing Standard Processing Subroutines	10-9
Processing Requirements for TSIDREL, TSIDLOC, and TSIDUNL.....	10-9
Processing Requirements for TSIDCHK	10-10
Executing in CICS.....	10-11

Using the Subroutines	10-11
Return Codes.....	10-12
Processing Requirements	10-14
Creating Job Setup/Run Sheets (TSIDFAV)	10-15
Chapter 11 Job Activity Reports	11-1
Recorder History Dataset.....	11-2
Creating the Activity Log Report (TSIDARP).....	11-3
EXCLUDE Control Statement	11-4
TSIDARP Control Statement.....	11-5
Creating the DSN/Job Cross-Reference Report (TSIDBRP)	11-6
Job Accounting Reports by Job (TSIDJAC).....	11-8
Job Accounting Reports by Partition (TSIDTAC).....	11-9
Maintaining the Job Accounting Database (TSIDJLD)	11-10
Online Access to the Job Accounting Database (TOJA).....	11-13
Chapter 12 Managing Tape Volumes	12-1
Introduction	12-2
Attention Routine Commands	12-5
FREE and STATUS Commands	12-6
Mounting Tapes	12-7
Output	12-7
Input.....	12-8
Initializing Tapes (TSIDINIT).....	12-9
Printing External Volume Serial Labels (TSIDLBL).....	12-14
Printing Dataset Labels (TSIDBPR and TSIDPRT)	12-16
Tape Vaulting.....	12-17
Introduction.....	12-17
Assigning Vaulting Methods to Datasets	12-19
Moving Volumes (TSIDVLT)	12-20
Reporting Current Locations of Vaulted Tapes	12-21
Manually Controlled Vaults	12-22
Vault To Dataset Cross-Reference Report (TSIDPRV).....	12-23
Dataset to Tape Report (TSIDDIS)	12-24
Tape to Dataset Report (TSIDSDS).....	12-25
Chapter 13 Disk Space Management.....	13-1
Introduction	13-2
Dataset Fragmentation	13-2
Secondary Allocation.....	13-3
Truncation.....	13-4
Converting Uncontrolled Disk Datasets.....	13-5
Placing Datasets on Specific Volumes in a Pool.....	13-6
VTOC Utilities	13-7
Introduction.....	13-7
Printing VTOC Entries (TSIDLVT)	13-8
Online VTOC Displays (TVTO).....	13-11
Offload System.....	13-15
Introduction.....	13-15
Selecting Datasets To Be Offloaded (TSIDOFD).....	13-17
Offloading and Backing Up Datasets (TSIDOFD).....	13-22
Onloading and Restoring Datasets (TSIDONL)	13-27

TSIDONL Control Statements.....	13-29
Differences Between TSIDONL Onload and TSIDONL Restore	13-32
Consolidating Offload Versions (TSIDOFD)	13-34
Appendix A BIM-EPIC Report Fields and Examples	A-1
Customized Reports	A-2
TOJA Report Examples	A-3
TSIDARP Report.....	A-6
TSIDBRP Reports	A-8
TSIDDIS Report	A-11
TSIDJAC Report	A-13
TSIDLVT Report.....	A-15
TSIDMNT Reports	A-17
TSIDOFD Report	A-19
TSIDOFL Report.....	A-20
TSIDONL Report	A-21
TSIDPRV Report.....	A-22
TSIDSDS Report	A-23
TSIDTAC Report	A-24
TSIDUTL Reports	A-26
Volume Serial Number and Vault Sequences	A-29
Display Vault Report	A-32
Display Scratch Report	A-33
Projected Scratch (PSCRATCH) Report	A-34
TSIDVLT Reports	A-35
Master Index	M-1

Trademark Information

This manual refers to the following brand or product names, registered trademarks, and trademarks which are listed according to their respective owners.

Software Diversified Systems, Inc.

DUSP™
MAXBACK/VSE®

Computer Associates International, Inc. (CA)

CA-ALERT®
CA-DRIVER
CA-DYNAM®
CA-DYNAM®/D
CA-DYNAM®/FI
CA-DYNAM®/T
CA-Easytrieve®
CA-EPIC®
CA-FAQS®
CA-FAQS®/ASO for VSE
CA-FAQS™/PCS
CA-JARS
CA-JCL/MANAGER®
CA-Librarian®
CA-LIBRARIAN®
CA-PANVALET®
CA-SORT
CA-SPACE/MANAGER®
CA-Super-EPAT®
CA-System/Manager®
CA-TAPE/MANAGER®
CA-VOLLIE®

Compuware Corporation

Abend- AID®

International Business Machines Corporation (IBM®)

CMS
IBM®
MVS
MVS/ESA™
MVS/SP®
MVS/XA™
VM
VM/CMS
VM/ESA®
VM/SP
VM/XA™
VSE
VSE/ESA®
VTAM®

Memorex Telex Corporation

Automatic Tape Library (ATL)
Library Management Software (LMS)

Storage Technology Corporation (Storage Tek®)
Host Software Component (HSC)

Software Pursuits, Inc.

MVT/VSE®
SPRI®

Syncsort Inc.

SyncSort®

Universal Software, Inc.

ADAS™
PSAM™

Preface

Most BIM-EPIC features operate only in conjunction with controlled datasets (datasets that BIM-EPIC catalogs and maintains), and controlled datasets are assumed in the discussion of each feature. Activity involving uncontrolled datasets is noted explicitly.

BIM-EPIC configuration options govern how BIM-EPIC operates and what you can expect BIM-EPIC to do. Default options are assumed in the main body of this manual. Variations caused by non-default options are described in the *BIM-EPIC Installation and System Guide*. Please consult your systems programmer if BIM-EPIC appears different from the description in the documentation.

This manual contains references to the products BIM-GSS and BIM-FAQS/ASO. Unless specifically stated otherwise, the products CA-GSS and/or CA-FAQS/ASO from Computer Associates can be used instead of the BIM products.

Special Terms

The following terms have specific meanings in this manual:

Term	Means...
Enter	Type the boldfaced characters and press the ENTER key.
Type	Type the boldfaced characters without pressing ENTER.
Character	Alphanumeric characters such as A to Z and 0 to 9.

Command Syntax

Descriptions of BIM-EPIC commands and control statements include the correct syntax or format used. The following example shows the special characters and conventions used to represent command syntax.

```
COMMAND  value
          [ ,PARM1=(value1,value2,...valuen)]
          { ,NOFUNCTION
            { ,FUNCTION  }
          [ ,DOTHIS=YES|NO ]
```

Upper-case characters (COMMAND) must be specified as shown.

Italics (*value*) represent variable information. In command lines, do not enter the characters shown. Instead, enter the information they represent.

In command lines, square brackets ([PARM1=...]) indicate optional items. To use the option, specify only the information inside the brackets. Do not code the brackets ([]) themselves.

Parentheses (()) should be specified as shown. They are used to group multiple subparameters or values together within a single parameter. If only one value is coded for a parameter, parentheses are not required.

Braces ({ NOFUNCTION }) also indicate required items. They surround lists of values or parameters where only one must be supplied. Specify only one of the choices. Do not code the braces ({ }) themselves.

The vertical bar symbol (YES|NO) separates choices that are mutually exclusive. Enter only one of the choices.

An underlined item (YES) indicates a default value that is used if the parameter is not specified.

Unless otherwise noted, parameters must be specified in the order shown.

Parameter Tables

Most commands, control statements, and macro statements use parameters. Parameter tables provide quick reference to those specifications. These tables indicate whether the parameter is required or optional, what values are valid, and any defaults. Parameter tables display information in the following way:

Parameter	Required?	Valid Entry	Default
<i>value</i>	Yes	1 to 99 characters	
PARM1	No	See syntax	VALUEA
One of the following two is required:			
FUNCTION			
NOFUNCTION			
DOTHIS	No	YES or NO	YES

Contacting BIM-EPIC Technical Support

You can reach BIM-EPIC Technical Support at any time. Our normal operating hours are from 8:00 a.m. to 7:00 p.m. Central Time. In addition to our regular customer support, we have technicians on call 24 hours a day, 7 days a week for emergency support. If you are outside the United States or Canada, contact your BIM Sales Agent.

Technical Support During Normal Operating Hours

Please remember that we rely on your description of the situation in order for us to determine its cause accurately. The more information you can provide, the more promptly we are able to solve your problem. By following the guidelines below, you can help us provide you with the exceptional service our customers have learned to expect.

Procedure

Step	Action
1	<p>Gain as thorough an understanding of the situation as you can, and classify your call using one of the following categories:</p> <ol style="list-style-type: none">1. MALFUNCTION IN A USER PROGRAM: A user program behaves differently with BIM-EPIC turned on than with BIM-EPIC turned off.2. MALFUNCTION IN VSE: VSE malfunctions when BIM-EPIC is turned on. For example, you receive non-interruptible wait states, system loops, and so on.3. BIM-EPIC FAILURE: An overall BIM-EPIC failure (abend).4. OPERATIONAL MALFUNCTION: A particular BIM-EPIC function or message does not operate correctly.5. QUESTION: Seeking “how-to” information or general knowledge.
2	<p>Have your documentation ready and available. We require the following documentation for problems in classes 1 through 4:</p> <p>A current TSIDDEB Summary Report, containing details concerning BIM-EPIC and your system configuration.</p> <p>The CONSOLE LOG (with our messages turned on): This provides a step-by-step account of the situation.</p> <p>The JCL, including expanded procedures: This is necessary to enable the technical staff to re-create the problem.</p> <p>A DUMP (if one is produced): This is necessary if we cannot re-create your problem due to differences in hardware or software.</p>
3	<p>Call us at 952- 933-2885. Please answer the questions asked by our receptionist as accurately as possible. This will enhance our ability to interpret your situation accurately and respond promptly.</p> <p>Our FAX number is 952- 933-7764.</p>

After-Hours Emergency Support

Please follow the guidelines outlined under "Technical Support During Normal Operating Hours" to gather the information required by our technical staff to solve your problem.

When calling for after-hours emergency support, an outside answering service will answer your call. Please be clear and precise when giving your name, company, your (after-hours) telephone number, and a brief description of the problem. Be sure to mention clearly that you need assistance with the product BIM-EPIC. EPIC can sound a lot like EDIT, another BIM product, and you could end up in the wrong support area.

The answering service will page the support technician on call. It will take approximately 35 minutes for you to receive a call back. If you do not receive a call in this time, call again to verify that the answering service has your correct phone number. Long delays usually indicate that we are having difficulty contacting you with the information given to us.

Related Publications

The documents described below contain important BIM-EPIC technical information. Familiarity with the relevant IBM operating system is assumed.

- *BIM-EPIC Installation and System Guide* contains detailed procedures for BIM-EPIC system installation, configuration, and maintenance. It assumes that you have access to the User Reference.
- *BIM-EPIC User Reference Guid* serves as both user guide and reference to BIM-EPIC operating features.
- *BIM-EPIC Messages and Codes* lists and describes BIM-EPIC messages and codes, including message responses and problem resolution.
- *BIM-EPIC Conversion Guide* contains instructions for converting to BIM-EPIC from another tape and disk management system.
- *BIM-EPIC Technical Release Notice* lists enhancements and feature changes that have been put in place since the last release. It also describes any special information required for upgrading to the current release.
- *Technical Notes*, distributed on the BIM-EPIC installation tape, contains the latest technical information, including updates to BIM-EPIC printed documentation.

1

Introduction

This chapter contains an overview of features and functions provided by BIM-EPIC®. This manual is written to include the novice mainframe programmer. If you have more experience in the field, you already are familiar with some of the subjects discussed and you may want to skim those sections. However, most of the chapter presents information unique to BIM-EPIC, so we encourage everyone to read the chapter for a full introduction.

Chapter Contents

BIM-EPIC Fundamentals.....	1-2
BIM-EPIC Dataset Definitions (EDDs)	1-9
Version Information	1-16
The Recorder File	1-27
Additional BIM-EPIC Features and Functions	1-28
The BIM-EPIC Environment	1-34
BIM-EPIC Job Control	1-39

BIM-EPIC Fundamentals

Configuration Options

Description

BIM-EPIC provides configuration options which allow you to customize many aspects of your processing.

Many configuration options are discussed in this manual. Please consult your systems programmer or the *BIM-EPIC Installation and System Guide* if you have a question about a specific configuration option.

Controlled Datasets

Description

BIM-EPIC controls datasets best when they are known to its **Dataset Name (DSN) Catalog**. The information contained in the records that make up a **DSN Catalog entry** determines how a dataset is processed.

With DSN Catalog control of a dataset, BIM-EPIC specifies the following:

- where each existing version is written and where new versions can be written
- how each version is written (allocation size, block size, density, multi-dataset, and so on)
- how long each version is retained and how many versions are kept at one time
- processing statistics for each version (creation date, last access date, block count)
- each tape version's vault location

Start Track 1 Datasets

Description

A **Start Track 1 dataset** is a disk dataset that is defined by the use of the number “1” in the relative track field of the VSE EXTENT statement. For example:

```
// DLBL MYFILE, 'ANY.FILE'  
// EXTENT SYS005,SYSWK3,1,0,1,20
```

The use of the number “1” in the relative track field above indicates that BIM-EPIC will allocate the dataset. The use of any other number in this field indicates that BIM-EPIC does not handle the allocation.

Start Track 1 datasets are only partially controlled. BIM-EPIC allocates disk space for the dataset on both input and output. However, you supply the name of disk pool to be used and the number of tracks or blocks to allocate for the dataset. All information is supplied by JCL.

Start Track 1 datasets provide one benefit: the dataset does not have to be entered in the DSN Catalog. However, they are less efficient than DSN Catalog datasets and they provide limited control over retention.

Start Track 1 datasets also require more I/O operations when they are opened. Because a Start Track 1 dataset's extents are not recorded in the catalog, BIM-EPIC must find the dataset's extents from the VTOC. In addition, all controlled datasets are written in pools, and pools can contain more than one volume. BIM-EPIC must read the VTOC of every volume in the pool until it finds the dataset's extents. This is true for both input and output. BIM-EPIC deletes any previous version of a Start Track 1 dataset before creating a new one. BIM-EPIC must search on output to locate and delete the old version.

In contrast, BIM-EPIC only needs to read a few records in the DSN Catalog to locate each extent of a cataloged dataset, no matter where it is written.

Start Track 1 datasets are retained and scratched like uncontrolled datasets. BIM-EPIC does not protect them from being scratched, and only one version can exist at a time. To retain the dataset using JCL, specify an expiration date or the number of days. If you do not specify an expiration date or Days Retention, BIM-EPIC provides a default retention of seven days.

Because of the limitations on retention, Start Track 1 datasets are useful only for work datasets. Because of the overhead required for these datasets, even a work dataset in a frequently run job should be cataloged to improve performance.

Configuration Options

Start Track 1 datasets are affected by the AUTOCAT, CATRT1, MINRET, NCATDSK, and STRTRK configuration options. In addition, they can be affected by other configuration options that apply to disk. See the *BIM-EPIC Installation and System Guide* for further information.

Controlling Different Types of Datasets

Description

BIM-EPIC can control SAM, ISAM, and DA datasets.

If the VSAM configuration option is set to *YES*, BIM-EPIC logs VSAM extent information in the DSN Catalog. Logging occurs each time a VSAM dataset is opened. BIM-EPIC always reflects the currently used extents, even if the dataset contains multiple secondary allocations.

BIM-EPIC can only control TYPEFLE=WORK files on disk. It cannot control them on tape.

BIM-EPIC does not control datasets processed with the DTF names IJSYSNR, IJSYSPC, or IJSYSRx, even if the dataset name is cataloged. The system programmer can exclude more DTF names from BIM-EPIC control by adding the DTF name to the ILIST parameter of the TSIDMAP macro.

Configuration Options

The CNTRL, UNCDISK, UNCTAP, and UNCTL configuration options affect the way uncontrolled datasets are processed. See the *BIM-EPIC Installation and System Guide* for details about the TSIDMAP macro.

DSN Catalog Entries

Description

BIM-EPIC's control of datasets is based on information contained in the DSN Catalog. All information pertaining to one cataloged dataset name is called a **DSN Catalog entry**. The format and contents of an entry are described below.

DSN Catalog information is divided into two levels: the **BIM-EPIC dataset definition (EDD)** level and the **version** level.

Each EDD contains all the attributes associated with a unique dataset name. When you process a controlled dataset, BIM-EPIC uses these attributes to control processing. The EDD is discussed later in this chapter.

BIM-EPIC enters version information automatically as each version is created. At least one catalog record is created for each version of the dataset. If the version spans more than one volume or extent, a catalog record is created for each additional volume or extent.

Version information include the version's location, its expiration date, creation date, and so on. BIM-EPIC uses this data to control the use and retention of each version individually.

The following sections give a detailed explanation of the data contained in each DSN Catalog entry, and how the various fields operate in conjunction with each other.

DSN Catalog Management

Description

Most updates to the DSN Catalog are made automatically as datasets are processed. However, sometimes you want to add a special dataset or change the defaults you have supplied for an existing EDD. This **manual cataloging** can be done using either the Online Manager or the TSIDMNT utility.

The BIM-EPIC Online Manager (EOM) is a panel-driven catalog management program. It leads you step-by-step through the catalog management process and allows you to view the modifications you have made to a dataset entry. The TSIDMNT program provides the same functions in batch mode.

Please refer to Chapters 2 and 3 for more information about what these programs can do.

DSN Catalog Maintenance

Description

The DSN Catalog must be maintained regularly. Maintenance includes regular backups and scratch runs, as well as periodic re-indexing.

Catalog maintenance is done using the DSN Catalog maintenance utility (TSIDUTL). See the *BIM-EPIC Installation and System Guide* for information about required maintenance.

BIM-EPIC Dataset Definitions (EDDs)

Introduction

Dataset Name

BIM-EPIC uses the dataset name in the DSN Catalog to identify a dataset as controlled. Each time a controlled dataset is opened, BIM-EPIC participates in processing.

The rules for dataset naming are:

- Dataset names can be up to 44 characters long, except for generation datasets (see below), whose names are limited to 35 characters.

Special considerations apply to tape datasets that belong to a multi-dataset group. See “Prime Dataset Name” under “EDD Tape Information” in this chapter.

Comment

You can record one 30-character comment with any BIM-EPIC dataset definition. Comments provide additional documentation for the dataset and are printed in BIM-EPIC reports.

Passwords

Passwords provide protection against unauthorized use of RELEASE and UNLOCK on this dataset and its versions. RELEASE and UNLOCK are described in Chapters 2 and 3. Entering a password does not lock, release, or unlock the dataset.

Encryption Password

Data encryption encodes tape datasets as they are written. They are decoded only when they are read under the control of the same DSN Catalog that was used to write them. Data encryption can be used to prevent tape datasets from being read at external locations. It provides no protection against unauthorized internal use.

Entering an encryption password for a dataset invokes data encryption for each tape version created. You do not have to remember this password.

The encryption password and the dataset password (above) function independently of each other. You can use both if you wish.

WARNING!

Encrypted dataset versions must not be deleted from the DSN Catalog. Once a version is deleted, it cannot be decoded and is unreadable. Furthermore the encryption password cannot be added, changed, or deleted once the dataset is cataloged.

User ID and System ID

System ID and User ID assist in grouping tape datasets for reporting and scratching purposes. Each can contain any two-character code that is significant to you and your installation. There

is no difference in function between them. You can use both together to group datasets at two levels. Either one or both can be used for any dataset.

SYS Number

You can cause a specific SYS number to be used for a dataset by entering that SYS number in the EDD. This inhibits LUB allocation (described in Chapter 7) for the dataset each time it is processed. We recommend that you avoid assigning a SYS number here unless it is absolutely necessary. It is better to allow LUB allocation to assign a SYS number for you, or to inhibit LUB allocation through JCL. If you do assign a SYS number in the EDD, ensure that you do not use the same number in your JCL for a different dataset in the same job step.

Block Size

Block size reblocks datasets. It overrides any block size coded in the program's DTF. If the block size specified is not a multiple of the record length, BIM-EPIC automatically rounds down to the closest multiple.

Reblocking is valid only for sequential datasets with fixed or variable record formats, and the program DTF must already specify a blocked format. It is supported for all datasets processed by logical IOCS, but may be ignored by programs that use physical IOCS to process the dataset. It is not valid for tape datasets which are read backward.

Block size and CISIZE are mutually exclusive; you can only specify one of these in a BIM-EPIC dataset definition.

Generation or Work Dataset

BIM-EPIC provides four retention criteria: **generation or work** (described here), and **retention days**, **cycles retention**, and **DLA retention** (described below). All of them work together to determine retention for active versions.

The 'generation or work' field specifies whether to keep more than one version of the dataset at a time. When you designate a dataset as a generation dataset, BIM-EPIC maintains *at least* the number of versions specified in the cycles retention field. Because each version is also retained as long as the retention days and DLA retention fields dictate, BIM-EPIC can retain more versions than the number specified with cycles retention.

When you designate a dataset as a non-generation (or work) dataset, BIM-EPIC retains only one version of the dataset at a time. As long as no new version is created, that version is retained as long as the retention days and DLA retention dictate. So, for disk versions, when you create a new version of a non-generation dataset, the old version is scratched automatically, whether or not the old version has expired. In addition, entries are automatically scratched from the VTOC to avoid the "OVERLAP ON UNEXPIRED FILE" error generated by VSE. For tape versions, the old version is scratched when you execute the SCRATCH function of the TSIDUTL program.

When you specify a cycles retention of one (1) for a non-generation dataset, BIM-EPIC always retains one version, even if its other retention criteria have expired.

Retention Days

The retention days field specifies the number of days each version should be retained after its creation date.

You can use the retention days field to specify permanent retention (PERM). When a disk version is created with permanent retention, an expiration date of 2099366 is recorded in the VTOC entry. When BIM-EPIC sees the 2099366 expiration date, it does not allow the operator to display or delete the dataset in response to an INSUFFICIENT SPACE condition.

Cycles Retention

The cycles retention field specifies the minimum number of versions (cycles) to maintain at all times. To maintain more than one version, you must also designate the dataset as a generation dataset (see above).

Days after Last Access (DLA) Retention

DLA retention relates expiration to dataset use. Even if other retention criteria have expired, the dataset cannot expire if it has been processed as input during the last n days.

EDD Tape Information

Standard Label	<p>The label field specifies whether an output tape dataset is to be created on a labeled or unlabeled tape.</p>
Tape Density	<p>The tape density specifies which density to use when creating a tape dataset. When a density is specified, BIM-EPIC only scans drives that support the specified density when it looks for a valid scratch tape. For example, you can use this field to assign a dataset to cartridge tapes only.</p> <p>A default density for your site can be specified using the MODE configuration option. Specifying a tape density in the EDD overrides the MODE default. Specifying a density in the JCL overrides both the MODE default and the EDD tape density. If no density is specified in any of these places, BIM-EPIC uses the last density that was assigned to the selected drive.</p>
Prime Dataset Name	<p>The prime dataset name links all datasets in a multi-dataset group. The link is accomplished in the DSN Catalog records, and it allows BIM-EPIC to group the datasets automatically. The prime dataset is the first dataset on the tape. The second and subsequent datasets on the tape are called subdatasets.</p> <p>Multi-dataset processing can be complex, and there are pre-processing tasks required before a multi-dataset group can be created. Multi-dataset processing is described in detail in Chapter 7.</p>
Tape Pool ID	<p>The tape pool ID field associates a dataset with a particular tape pool. Once a dataset is assigned to a tape pool, it can only be written on scratch tapes from that pool. Datasets not assigned to a tape pool can only be written on scratch tapes that are not assigned to any tape pool.</p>
Owner ID	<p>The owner ID is a secondary pooling mechanism for tape datasets. An owner ID is a two-character code that matches the first two characters of a tape's volume serial number. Owner ID is provided primarily for compatibility with older tape management systems. If an owner ID is specified for a dataset, all new tape versions must be created on tape volumes which begin with those two characters.</p>
Revolving Tape Datasets	<p>Specifying Revolving Tapes=YES for a dataset establishes an exclusive relationship between the dataset and the tapes it uses. A revolving tape volume can only contain a version of a specific dataset, and a revolving tape dataset can only be written on specific tape volumes. A revolving tape volume cannot be used in an Automated Cartridge Library (ACL).</p>

Tape Size	If you specify Tape Size as SMALL, BIM-EPIC requests a small scratch tape when the dataset is created. If a tape volume serial number ends in ‘S’, BIM-EPIC considers the tape small. Large tapes are the default, and they have no restriction on their volume serial numbers.
Vault Method Number	A vault method establishes a pattern of movement through tape vaults for each tape volume in the dataset. The vault method number field assigns a vault method and activates vaulting for the dataset.

EDD Disk Information

CISIZE

The CISIZE field specifies a CISIZE for FBA devices and forces the dataset to be written to maximize the number of records written to a control interval. Use CISIZE instead of block size for datasets to be written on FBA devices.

Block size and CISIZE are mutually exclusive; you can only specify one of these in a BIM-EPIC dataset definition.

Disk Pool Name

The disk pool name field is used to assign disk datasets to a disk pool. Disk pools are created by the systems programmer. They are referenced by a six-character pool name which can be a real volume serial number or an assigned name.

The systems programmer can establish CPU or partition-independent disk pools. To specify CPU and partition-independent pools, use three dashes at the end of the pool name (for example, POL - - -). For partition-independent pools, use two dashes at the end of the name (for example, POOL - -).

If you don't assign a pool, and a default pool has been specified by the DEFPOL configuration option, the dataset will be assigned to the default pool.

Primary Extent Size

The primary extent size field defines the number of tracks or blocks to be allocated to the first extent. The minimum size is 1 CKD track or your CISIZE for FBA.

Secondary Extent Size

Secondary extents are allocated automatically to a version of a sequential disk dataset when the primary extent is filled. Secondary extents are only supported for programs that use logical IOCS and programs that use physical IOCS programs if they support secondary allocations.

The secondary extent size field defines the number of tracks or blocks to be allocated to every extent after the first. The default secondary extent size is 50% of the size of the primary extent.

Number of Secondary Allocations

You can specify the maximum number of secondary extents allowable for a single dataset version. The maximum is 255.

Allocation by Logical Records

Allocation by logical records (ALR) is used to calculate the size of the primary extent automatically. It overrides the primary extent size field. BIM-EPIC calculates the extent size based on:

- record length

- estimated number of records
- block size
- track/block capacity of the selected volume.

ALR allows you to migrate from one device type to another without changing the allocation size.

ALR is valid only for sequential disk datasets created with a DTFSD.

Version Information

Introduction

Description

As each version of a dataset is created, BIM-EPIC automatically records attributes specific to that version in the DSN Catalog. The attributes recorded at this level are described in this section.

Version Number

The current (most recently created) version is always version 1. Whenever a new version of a dataset is created, previous versions are incremented automatically. Versions in scratch, open, or conditionally cataloged status (see below) do not have version numbers.

Generation Number

A generation number is assigned to each generation disk version. It is a nine-character suffix that is appended to the dataset name (in the VTOC only) as “.G=nnnnnn”, beginning in position 36 of the dataset name.

Since several versions of a generation dataset can be written on the same disk volume, BIM-EPIC appends a unique generation number to the dataset name to ensure that the dataset name in the VTOC is unique.

Although the VSE VTOC name of the dataset includes the generation number, BIM-EPIC does not consider the generation number to be part of the dataset name. Therefore, do not include it in your JCL when you want BIM-EPIC to process the dataset under DSN Catalog control. Specify the dataset name only as it appears in the BIM-EPIC dataset definition.

Version Status

BIM-EPIC tracks the status of each version. BIM-EPIC status types are:

Active (A) is the default status of an unlocked active version.

Open (O) indicates that close processing did not complete successfully. An open version cannot be used or scratched until you declare it closed using the Online Manager or TSIDMNT program.

Conditional Catalog (C) indicates that the version was created with the TESTJOB feature or a conditional catalog option in the JCL. It is accessible as input only if it is the latest in the dataset’s chain of versions. In addition, it can only be accessed by a job with the same name as the job that created it. To make it an active version, declare it closed it using Online Manager or the TSIDMNT utility.

Open and conditionally cataloged versions have different rules for retention from active versions. See “Rules for Dataset Retention” in this chapter for details.

Scratch (S) denotes a tape version in scratch status. The tape can be overwritten at any time. Disk version records are deleted, not marked as scratch.

Lock (L) denotes a version in locked status. It cannot be accessed as input until it is unlocked or released using Online Manager or the TSIDMNT utility.

Subdataset (N) denotes a dataset in a multi-dataset group with file sequence number 2 or greater.

Work (W) indicates a non-generation dataset.

Unlabeled (U) indicates that the version was created on an unlabeled tape.

MCAT (M) indicates that the version has been MCATALOGed. It can be an MCAT master or alias version.

Offload status indicates that a disk version has been moved to tape with BIM-EPIC's offload system. The offload system is described in detail in Chapter 13.

Volume Serial Number

The volume serial number field indicates the volume serial number of the tape or disk volume on which the version is written. For multi-volume datasets, there is one volume serial number recorded for each volume.

Reel Number

The reel number indicates the tape volume sequence number for multi-volume datasets.

File Sequence Number

The file sequence number field indicates the position of the dataset in a multi-dataset group.

Extent Number and Address

BIM-EPIC records the extent addresses used for each disk dataset. For each extent it records:

- the extent sequence (1 for primary, 2 for the first secondary, and so on)
- the beginning relative track or block number
- the size of the initial allocation
- the amount of allocation actually used

The amount of allocation actually used is smaller than the initial allocation if the dataset has been truncated.

BIM-EPIC can allocate an extra extent (called an EOF extent) if the extent being processed is nearly full when the dataset is closed. See 1-43 in this manual.

Record Format, Length, Block Count, Block Size

The version's actual record format, record length, block count and block size are recorded. This information is recorded when the dataset is created by a program that uses logical IOCS; it can be zeros if the dataset is created by a program that uses physical IOCS.

Creation Date/Time	BIM-EPIC records the date and time the version was written.
Creating Job/Device Address	BIM-EPIC records the name of the job that created the version, the partition in which it was created, and the address (cuu) of the device on which it was created.
Last Access: Job and Date	At each open, BIM-EPIC records the date the version was accessed and the name of the job that accessed it.
Expiration Date	BIM-EPIC calculates an expiration date for each version based on retention days and DLA retention. A version remains active past its expiration date if it is still needed to satisfy cycles retention.
Tape Use and Error Count	<p>The tape use count indicates the number of times the tape has been opened. The tape error count indicates the number of read/write errors encountered.</p> <p>Both counts are kept for each tape while it has belonged to this version and for total tape use since the last date the tape was cleaned.</p>
Current Vault and Slot Location	The current vault and slot location are recorded for each tape if the dataset is subject to a vault rotation method.

Automatic Cataloging

Description

Each new dataset and version must have an entry in the DSN Catalog to be controlled. These entries can be created automatically for uncontrolled datasets using the AUTOCAT and AUTOLOG features or the parameter DISP=(x,CATLG) parameter on a DD statement.

AUTOCAT and AUTOLOG are configuration options described in the *BIM-EPIC Installation and System Guide*.

AUTOCAT and AUTOLOG operate globally on all qualifying datasets. DISP=(x,CATLG) operates locally on the single dataset named in the DD statement that contains this parameter. For more information on DISP=(x,CATLG), see Chapter 6, “BIM-EPIC DD Statements”.

AUTOCAT and AUTOLOG are independent but parallel functions. They operate under the same rules. However there is an important functional difference between them: AUTOCAT catalogs uncontrolled *output* tape datasets and versions. AUTOLOG catalogs uncontrolled *input* tape datasets only (no versions).

AUTOCAT

AUTOCAT creates a dataset definition for an output dataset if that dataset is not already entered in the DSN Catalog. All subsequent versions created under the dataset name will also be cataloged, including the one that triggers the AUTOCAT operation.

When autocataloging multi-dataset groups that are read out of sequence, ensure that the prime dataset is entered in the DSN Catalog before any subdataset in the group is read. This requirement is automatically satisfied when the datasets are read in sequence beginning with the prime dataset.

The following fields are entered in each autocataloged dataset definition:

- Retention days
- Cycles retention
- Default disk pool
- Default primary extent

On reports and displays, automatically cataloged datasets are identified by a User ID of “AC” and given the comment “***** AUTO CATALOGUE *****”. All automatically cataloged datasets are generation datasets. These dataset definitions can be changed later using Online Manager or the TSIDMNT utility.

AUTOLOG

AUTOLOG creates a BIM-EPIC dataset definition (EDD) for an input tape dataset that is not already in the DSN Catalog. However, AUTOLOG does not also autocatalog any versions to the EDD, as AUTOCAT does. AUTOLOG is primarily used to force standard EP001 MOUNT messages for previously uncontrolled input tape datasets.

When autologging multi-dataset groups that are read out of sequence, ensure that the prime dataset is entered in the DSN Catalog before any subdataset in the group is read. This requirement is satisfied automatically when datasets are read in sequence beginning with the prime dataset.

Configuration Options

Automatic cataloging is affected by the AUTOCAT, AUTOLOG, CATRT1, NCATDSK, CYCL, RETEN, DEFPOL, and DEFEXT configuration options. AUTOCAT functions can be specified differently for tape and disk.

There are three choices for automatic cataloging functions: Global, SYSPARM, and None.

Global AUTOCAT

Global automatic cataloging catalogs all datasets.

SYSPARM AUTOCAT

Only selected datasets are cataloged under SYSPARM automatic cataloging. They are cataloged automatically if the current SYSPARM matches the SYSPARM in the AUTOCAT or AUTOLOG configuration option.

No Automatic Cataloging

When you turn off automatic cataloging, new datasets can only be cataloged using Online Manager or the TSIDMNT utility.

Retention and Scratching

Description

All catalog-controlled datasets must be scratched before tapes and disk space can be reused. However, BIM-EPIC does not scratch any datasets unless specifically told to do so. This can be done in four ways.

- Using JCL options (work/non-generation datasets only)
- Defining a dataset as “work” or as a common work dataset (CWDS)
- Using the Online Manager or TSIDMNT
- Using the SCRATCH function of the TSIDUTL utility

The JCL scratch functions can be used to delete work (non-generation) dataset versions when they are closed as input or at the end of the job or job step. These versions are scratched regardless of whether or not the retention criteria have been satisfied.

By design, BIM-EPIC maintains only one version of a work (non-generation) dataset. As soon as a new version is opened, work dataset scratch processing immediately deletes any previous disk versions. Any previous tape versions become eligible for scratch.

BIM-EPIC common work datasets (CWDS) are retained only for the duration of a VSE job. Scratch processing always deletes a CWDS at end of job, whether the job completes normally or not.

Online Manager and TSIDMNT scratch functions delete only the dataset versions you specify. Those versions are scratched regardless of whether or not the retention criteria have been satisfied.

The SCRATCH function of the TSIDUTL program reviews all the datasets in the DSN Catalog and deletes all versions that are eligible for scratch. A dataset is considered eligible for scratch when ALL the retention criteria have been satisfied.

You can also preview a scratch run with TSIDUTL’s projected scratch function (PSCRATCH) which gives you a listing of all tape datasets that are eligible for scratch on a particular date. Use this report to identify any expired datasets you might want to save, and to obtain an estimate of the number of scratch tapes you will have available at any point.

Rules for Dataset Retention

Cataloged Datasets

BIM-EPIC's retention rules for cataloged datasets are:

Tapes are not made available for reuse and disk space is not released until dataset versions are scratched.

Non-temporary versions are not scratched until the SCRATCH function of the TSIDUTL utility is executed. TSIDUTL does not scratch any versions until they are eligible. A version is not eligible until ALL the retention criteria have been satisfied.

Any version can be scratched at any time using the SCRATCH function of Online Manager or the TSIDMNT utility.

A previous work dataset version is scratched immediately when a new version is opened. A previous tape work dataset version becomes eligible for scratch when a new version is opened.

Work dataset versions can be scratched at close (on input) or at end of job or job step using JCL options. Disk space is freed and tapes become available for reuse immediately.

Multi-dataset tape volumes are not made available for reuse until all dataset versions on the tape have been scratched.

Conditionally cataloged versions become eligible for scratch the day after their creation.

Common work dataset versions are always scratched at end of job.

Open versions become eligible for scratch according to the THLDOPEN and DHLDOPEN parameters of the TSIDUTL utility's SCRATCH function. The default is to retain open tape versions for one day and open disk versions for two days after their creation dates.

WARNING!

Tapes in open status can be reused before they are actually scratched. They are accepted as scratch tapes if they are accessed in the same partition with a different job name. While this means that the tape is vulnerable to accidental reuse, it also means that these tapes are immediately available as scratch. Please ensure that your operators are informed of this. Use of tapes in open status can be modified by BIM-EPIC special options.

Examples

Description

This section contains examples of retention under multiple criteria.

Retain x Versions for y Days

You need to retain each version of an inventory dataset for 100 days. You also need to retain the last 10 versions, regardless of when they were created. The dataset definition specifies:

```
Generation Dataset.....YES
Retention Days.....100
Retention Cycles.....10
Retention DLA.....0
```

Case 1

During the first 100 days after creating the first version, you create a total of nine versions. On day 101, no versions are scratched because you still have not completed 10 cycles.

Case 2

If you create Version 11 after only 60 days, no version is eligible for scratch because the first version you created is only 60 days old. It is retained until day 100 regardless of the number of versions created up to that time.

Retain a Work Dataset x Days after its Last Access

An accounting job produces results from work datasets in multiple stages. If you decide to save the work datasets for 15 days after the last time you use the data, catalog the dataset with:

```
Generation Dataset.....NO
Retention Days.....0
Retention Cycles.....0
Retention DLA.....15
```

Each version is saved for 15 days after its last access.

Temporary Datasets

If you want a temporary dataset, you specify:

```
Generation Dataset.....NO
Retention Days.....0
Retention Cycles.....0
Retention DLA.....0
```

All versions are eligible for scratch at all times.

Retention for Start Track 1 Datasets

Description	<p>By default, BIM-EPIC’s retains Start Track 1 datasets with period retention only. Expiration is determined by the JCL or by the default retention for Start Track 1 datasets specified in the DSKRET configuration option. The existing version is deleted as a new version is created.</p> <p>The space occupied by a Start Track 1 dataset becomes available when the expiration date in the VTOC has been reached.</p>
--------------------	--

Overriding Standard Retention

Description

Since the default retention criteria may not be suitable for every version of every dataset, you may want to modify retention criteria occasionally. You can change retention for new versions, for existing versions, or for the dataset as a whole.

Retention days or an expiration date specified on a TLBL, DLBL, or DD statement for an output dataset overrides the catalog default.

You can change the retention of a cataloged version with the RETAIN and CYCLE commands of Online Manager or the TSIDMNT program. To update the retention for all new versions of a dataset, you must update the retention of previous versions explicitly with RETAIN or CYCLE.

You can update existing Start Track 1 datasets to permanent retention with the RETAIN, PERM, and PACKSCAN functions of TSIDVUT. You can delete them before their expiration date is reached with the DELETE or PACKSCAN functions.

Automatic Generation Purge

Description

When **Automatic Generation Purge (AGP)** is enabled, BIM-EPIC automatically deletes DSN Catalog information about any disk dataset whose version number exceeds the cycles retention in the BIM-EPIC dataset definition (EDD).

AGP is enabled by setting the AGP configuration option to YES when creating or updating an EDD in the DSN Catalog.

What Happens if the Job Cancels?

If the configuration option PURBYP is set to 'YES', the automatic purge is bypassed if the job cancels. If PURBYP is set to NO, AGP remains in effect and the version is deleted.

AGP Example

The following TSIDMNT command adds an EDD to the DSN Catalog. BIM-EPIC retains 3 versions of the dataset *DSNA* at all times. When a new version is created, the oldest version is automatically deleted at end-of-job.

```
CAT    'DSNA',CYC=3,AGP=YES,...
```


The Recorder File

Description

The Recorder logs controlled dataset activity. This includes:

- all DSN Catalog maintenance transactions, excluding TSIDVLT (vaulting) maintenance
- all opens and closes of controlled datasets
- machine use and job accounting statistics

The Recorder File is vital to emergency recovery of the DSN Catalog. It contains a record of all changes made to the DSN Catalog since the last catalog backup. BIM-EPIC uses the data in the Recorder File to bring your recovered DSN Catalog completely up to date.

Recorder data can also be used to produce job accounting reports.

You can also log your own comments in the Recorder File for documentation using the Recorder maintenance utility (TSIDRFS) which is documented in the *BIM-EPIC Installation and System Guide*.

WARNING!

We strongly recommend keeping the Recorder active at all times.

Configuration Options

Configuration options RECORD, RECWARN, RECWMSG, and RECSHR affect the functioning of the Recorder feature.

Additional BIM-EPIC Features and Functions

LUB Allocation

Description

VSE's Logical Unit Block (LUB) connects the logical unit (SYS number) in a program's file definition (DTF) with the physical device (cuu) that controls the dataset.

The LUB allocation feature of BIM-EPIC assigns logical units for tape and disk devices if they are opened when the SYS number specified in the program is already in use. This eliminates potential conflicts in SYS number assignments when:

- a program or job stream uses the same SYS number for two different datasets, and the datasets are stored on two different DASDs or
- the requested SYS number is permanently assigned to another device

The LUB allocation feature reassigns new SYS numbers (LUBs) to both datasets, and continues processing without interruption. In addition, LUB allocation allows BIM-EPIC to write multiple extents for a single dataset on separate disk volumes to make full use of its disk pooling feature.

When a dataset is opened, if its SYS number is already in use, BIM-EPIC selects an unused SYS number. For disk datasets, it begins assigning SYS numbers with the number specified in the DSKLUB configuration option. For tape datasets, it begins with the number specified in the TAPLUB configuration option.

LUB allocation is automatic in most circumstances. In some cases, it is inhibited automatically. You can also inhibit it using JCL parameters.

Managing Tape Devices

Drive Assignments

BIM-EPIC makes and releases drive assignments automatically with the Automatic Volume Recognition (AVR), Autoattach, and Early Drive Release features. With these functions, tape drives are assigned to a partition or virtual machine only between open and close. They aren't tied up between job start and open time or between close time and EOJ, so tape drive availability is increased.

Tape drive assignments are affected by the SHARE and AUTOATT configuration options. Early Drive Release is activated for uncontrolled tapes if the configuration option UNCTAP=YES is specified.

Automatic Volume Recognition

BIM-EPIC does **Automatic Volume Recognition (AVR)** processing for both tape and disk. When necessary, it searches available drives for the correct volume and assign that drive to the partition running the job.

A tape drive is considered available for output if it:

- contains a valid tape, that is, the tape volume serial number is in the DSN Catalog in scratch status, or it has been initialized by DITTO
- is ready and at load point
- supports the mode setting selected for the dataset
- is not in “device down” status
- is not assigned to any partition
- is not attached to any VM machine if Autoattach is on, or is attached to your machine if Autoattach is off

When AVR finds the tape, BIM-EPIC assigns the drive to the partition running the job. To function at its best, AVR requires you to delete your ASSGN statements from your JCL. For more information about assignments and JCL, please see “ASSGN Statements and LUB Allocation”.

Auto Stop

If AVR cannot find an appropriate tape, BIM-EPIC pauses the partition and waits for the interval specified by the AUTOSTP configuration option, then AVR scans the drives again. This process repeats at the interval specified by the AUTOSTP configuration option. If AVR still cannot find the appropriate tape after this process has completed, then it sends a mount message to the operator and puts the partition in a wait state until the operator responds.

Autoattach

Autoattach attaches tape drives to virtual machines. Autoattach works with AVR to attach a tape drive to the virtual machine while AVR assigns it to the partition.

Autoattach adds two new commands to VM: VATTACH and VDETACH. VATTACH corresponds to the IBM ATTACH command. VDETACH corresponds to the IBM DETACH command.

BIM-EPIC issues these commands during AVR to search for the desired tape on a drive that is owned by VM.

These commands differ from their IBM counterparts in the following ways:

- VDETACH detaches the tape drive without unloading the tape. This leaves the tape accessible for another AVR search.
 - Both the VATTACH and VDETACH commands operate in “silent” mode. That is, they do not issue messages.
-

Early Drive Release

At close time, BIM-EPIC releases the tape drive assignment with **Early Drive Release** and Autoattach.

Early Drive Release and Autoattach are automatic, but they are inhibited if you use alternate assignments.

Autoattach cannot detach a drive if the job cancels or when the assignment is held past close with the TLBL option 8 or DD parameter FREE=END. However, another job on that virtual machine can be assigned to the drive, and if that job releases the assignment at close, Autoattach detaches the drive automatically.

Managing Your Tape Volumes

BIM-EPIC's tape initialization program (TSIDINIT) prevents initializing tapes with duplicate volume serial numbers and adds new tape volumes to the DSN Catalog.

TSIDLBL creates gummed volume labels for your tapes.

Vault management facilities control tape vaulting.

Tapes are added to the DSN Catalog automatically as you process datasets. In most circumstances, they do not have to be added manually. If you wish to add tapes to the DSN Catalog manually, use the ADD command (in Online Manager or the TSIDMNT program) or reinitialize the tapes with TSIDINIT.

You can migrate to cartridge tapes automatically by specifying a cartridge density as the global default for all output tapes. To do so, use the MODE configuration option.

Managing Disk Devices

Disk Pooling

BIM-EPIC uses disk pooling to make disk allocation more flexible and more efficient. The systems programmer defines disk pools for use in your installation. Disk pools can include one or more disk volumes. Part of a volume can also be allocated to a pool.

Each disk dataset must be assigned to a disk pool before allocation. Each time the dataset is written, its allocation is made in that pool. Disk pools are referenced with a six-character pool name. The same pool can also be referenced by alias pool names. In addition, you can have CPU and partition-independent pool names. If these are defined, you can reference a partition independent pool by ending its name with two dashes (for example, P O O L - -). For CPU and partition independent pools, end the name with three dashes (for example, 'P O L - - -').

Pool names are assigned to datasets through JCL or the DSN Catalog.

Leveling

Leveling reduces head contention by distributing allocations over the entire disk pool, rather than concentrating them on a single volume. Each time a new extent is written, BIM-EPIC makes the allocation on the next volume in the pool sequence.

Leveling is activated for each disk pool when the pool is defined.

Offload Utilities

BIM-EPIC's offload utility moves datasets from disk to tape and back again. Processing is done logically, so you can use the offload utility for backup, or to reorganize your disk space, or just to offload infrequently used datasets (which remain known to BIM-EPIC and automatically reload to a temporary disk dataset whenever needed). The only restriction is that the offload utility only processes controlled, sequential datasets.

VTOC Utilities

BIM-EPIC provides three utilities for displaying and updating your VTOC entries. TSIDLVT lists VTOC entries in batch mode. TVTO displays them online. TSIDVUT allows you to modify existing VTOC entries.

Starting and Stopping Partitions

Description

You can stop a partition by responding STOP to most BIM-EPIC messages. To restart, submit the following command to the attention routine (AR):

MSG *xx*

where *xx* specifies the partition identifier (for example, BG, F1, F2).

Partitions can be stopped and restarted at any time using the VSE STOP and START commands.

Creating Reports

Description

BIM-EPIC programs provide a variety of reports about tape and disk datasets, storage use, and even job accounting. For report examples and an explanation of report fields, see “Report Fields and Examples”.

You can create your own reports containing BIM-EPIC data from the DSN Catalog and the Recorder File. For more information, see “Customized Reports”.

The BIM-EPIC Environment

When is BIM-EPIC On?

Description

BIM-EPIC is activated by partition. If you have just IPLed your system, or you have deactivated and reactivated BIM-EPIC, BIM-EPIC becomes functional when the first OPEN is issued in the partition.

BIM-EPIC does no processing until it is activated. This means that JCL enhancements (such as DD statements) cannot be processed until then.

There is one exception to this rule: QDAS can always be active, whether BIM-EPIC is active or not. QDAS is described below.

What Happens When BIM-EPIC Scratches or Deletes a Dataset?

Description

BIM-EPIC provides several methods for scratching datasets. Datasets can be scratched at your request using JCL, Online Manager, the TSIDMNT catalog management utility, or the TSIDUTL catalog maintenance utility. Regardless of which method is used, BIM-EPIC scratch processing is the same.

When BIM-EPIC scratches a disk dataset, it deletes the VTOC entry, and that disk space is made available for reuse. The DSN Catalog entry for that version is deleted.

For tapes containing only one dataset, the tape is made eligible for scratch immediately. For multi-dataset tapes, the tape will not be scratched until all its datasets have been scratched. The dataset's version entry in the DSN Catalog is changed to scratch status; it is not deleted. If the dataset's definition (EDD) still exists, the scratched version record remains in the DSN Catalog. If the dataset definition is deleted, the tape version record is moved to the definition name

*** DSN DELETED ***.

For both tape and disk, only the version record is affected in normal scratch processing. However, dataset definitions can also be deleted from the DSN Catalog with the DELETE and PURGE functions. DELETE scratches all versions and VTOC entries but does not remove tape volumes from the DSN Catalog. PURGE scratches all versions and VTOC entries, and removes tape volumes from the DSN Catalog.

BIM-EPIC JCL

Description

BIM-EPIC changes the way you use JCL. It offers many enhancements to standard VSE job control. It also requires some changes to function properly.

Physical Input/Output Control System and Pre-Open Checking

Description

Programs that use physical IOCS cause conflicts with tape and disk management programs. As the following sections describe, BIM-EPIC doesn't get control until your program issues a request to the VSE supervisor for open, end-of-volume/extent, or close.

Physical IOCS programs cause conflicts because they don't strictly follow most logical IOCS program conventions. For example, they perform "pre-open checking": they check to see what kind of device is to be used to process a dataset before they open it, before BIM-EPIC can make the assignment. Most PIOCS programs don't update the DTF with block size, record size, record count and address of the last record written, so BIM-EPIC can't find the necessary information about the dataset at end-of-volume/extent or close.

However, processing controlled datasets with physical IOCS is not difficult. It requires some special definitions in the JCL and, in some instances, special defaults in the BIM-EPIC dataset definition (EDD). See 7-35 in this manual for complete requirements.

What happens if the job cancels?

Description

At open time, versions are entered into the DSN Catalog with an open status. When close processing is requested, BIM-EPIC updates the version record and indicates that the version has been closed and is now active.

If the job cancels before a version is closed, the version remains in open status. Open versions cannot be used as input, and they become eligible for scratch differently from closed versions. Generation datasets become eligible for scratch according to the time limits you set for the THLDOPEN and DHLDOPEN parameters of the TSIDUTL program's SCRATCH function. The default is to scratch open disk datasets after two days, tape datasets after one day. Non-generation datasets are automatically scratched the next time an output version is created. You can also use BIM-EPIC JCL options to specify that non-generation disk datasets are to be deleted immediately if a job cancels.

If a dataset is successfully closed before the job cancels, it becomes an active version, and is handled exactly as if the job had processed normally.

BIM-EPIC Job Control

Introduction

JCL Enhancements

Because BIM-EPIC offers JCL enhancements which VSE cannot process, BIM-EPIC processes JCL options before VSE so that no error is generated. BIM-EPIC checks dataset status and generates the required VSE label information.

Tape Processing

Open Processing

Open, end-of-volume, and close processing are the most significant processing periods for BIM-EPIC, and open processing is the most important of the three. In each case, processing is different for tape and disk. Each is discussed separately. When a tape dataset is opened, BIM-EPIC carries out the following tasks:

BIM-EPIC determines whether a dataset is tape or disk by checking the JCL and the DSN Catalog.

BIM-EPIC determines whether the dataset is controlled. If it isn't controlled, BIM-EPIC passes control to VSE immediately. If the dataset is controlled, it reads the catalog entry to determine how to process this version.

BIM-EPIC performs AVR processing by searching available drives for the correct input tape or for a BIM-EPIC scratch tape for output.

If it doesn't find an appropriate tape, it issues a message to the operator and waits for a response.

When the tape is mounted, it assigns the drive to the requesting partition and attaches the drive to the VSE machine if Autoattach is active.

It updates the DSN Catalog (creates a dataset definition if automatic cataloging is required and records information about the version being created). The version is placed in open status.

BIM-EPIC updates the VSE Label Area. On input, it clears the dataset name from the label and substitutes the correct volume serial number. On output, it adds a period to the VOL1 label to mark the tape as a BIM-EPIC tape and writes a 99365 expiration in the HDR1 label to protect the tape against accidental overwrites.

BIM-EPIC passes control to VSE before the dataset is written. VSE then uses the information BIM-EPIC has written in the VSE Label Area to process the dataset.

End-of-Volume Processing

The steps performed in end-of-volume (EOV) processing are similar to those performed in open processing. BIM-EPIC sets the stage for reading or writing the next tape. In EOV processing, BIM-EPIC must gain and pass control to VSE more than once.

BIM-EPIC releases the assignment made for the drive that was just used and detaches the tape drive from the VSE machine if Autoattach is active.

It performs AVR for the next volume.

If it can't locate the volume, it issues a message to the operator.

When the tape is mounted on an available drive, BIM-EPIC assigns and attaches the drive.

It updates the DSN Catalog. On output, it writes another volume record for the version, places that volume in open status, and places the previous volume in conditional catalog status. On input, it updates date last accessed, use count, and other statistical information.

It passes final control to VSE.

**Closing Tape
Datasets**

In close processing for tape, BIM-EPIC gains control from and passes control to VSE more than once.

BIM-EPIC tells VSE how to position the tape, whether to unload it, rewind it, or leave it in place.

It releases the drive's assignments and detaches the drive from the VSE machine if Autoattach is active.

On output, it writes to the DSN Catalog, placing all volumes in closed status.

BIM-EPIC passes final control to VSE.

Disk Processing

Open and End-of-Extent Processing

The tasks required for disk open and end-of-extent processing are similar to those for tape. BIM-EPIC reads the DSN Catalog, issues messages, makes an assignment, and so on. For disk, however, BIM-EPIC must make the allocation, which is more complicated than finding the correct tape volume.

Quick Disk Allocation (QDAS)

At every open, VSE reads the VTOC sequentially, which can require thousands of I/O operations. BIM-EPIC provides the **Quick Disk Allocation System (QDAS)** to dramatically decrease the time required to allocate an output dataset.

QDAS maps disk volume free space to virtual storage. One map is created for each volume when the system is initialized. When disk space is used or freed for any dataset (controlled or uncontrolled), QDAS updates its map in storage.

Operational Notes: QDAS only updates its maps when the VTOCs of the disk volumes under its control are updated using the VSE common VTOC handler (\$IJHCVH). Programs that update the VTOC directly may bypass the QDAS interface and cause its maps to get out of sync with the VTOC. This applies to a very limited number of programs (such as DITTO). However, should this condition occur, you can refresh the QDAS map tables at any time. See the *BIM-EPIC Installation and System Guide* for procedures.

Because QDAS only maps free space, it is necessary to refresh its tables daily to pick up the space made available by Start Track 1 files that have expired. This is not necessary if all your disk files are catalog controlled. It is recommended that this procedure be performed daily after midnight as part of the BIM-EPIC daily maintenance procedures. See “Regular Maintenance” in Chapter 4 of the *BIM-EPIC Installation and System Guide* for further information.

On Output

1. BIM-EPIC locates the pool to which the dataset is assigned. For cataloged datasets, it finds the pool in the DSN Catalog entry. For Start Track 1 datasets, it finds the pool that is specified in the JCL.
2. If the dataset is non-generation, BIM-EPIC deletes the old version. This is simple for cataloged datasets, since the location of the dataset is contained in the DSN Catalog. For Start Track 1 datasets, BIM-EPIC has to read each VTOC in the pool until it finds the old dataset.
3. BIM-EPIC decides which disk volume in that pool will be allocated using the leveling feature, if active, so that datasets will be allocated evenly across all volumes in the pool.
4. BIM-EPIC checks the VTOC (or QDAS) for a large enough free extent. (BIM-EPIC does not find free extents through the DSN Catalog.) If it cannot find sufficient space in the pool for the extent, it issues a message and waits for a response from the operator.
5. BIM-EPIC updates the VSE Label Area, creating an extent and assigning the disk volume.

On Input

1. BIM-EPIC locates the dataset. For cataloged datasets, it finds the dataset's location in the DSN Catalog. For Start Track 1 datasets, it has to read each VTOC in the pool until it finds the first extent of the dataset.
2. It updates the VSE Label Area, creating an extent and assigning the disk volume.

Cataloged dataset processing requires a great deal less overhead than Start Track 1 processing. Although BIM-EPIC has to write to the DSN Catalog each time it accesses a cataloged dataset, it uses many fewer I/O operations than would be required to read all the VTOC entries found in your disk pool. We strongly recommend using cataloged datasets instead of Start Track 1 datasets in most cases. We also recommend that you use Start Track 1 datasets only in disk pools that contain fewer than three volumes.

Space Holder Records

Before we can discuss close processing for disk datasets, you need to understand **space holder records**.

Each time BIM-EPIC allocates an extent for a dataset, it writes a VTOC entry called a space holder record to prevent the same space from being used when allocating another BIM-EPIC dataset. The VTOC entry for the space holder record contains the name of the dataset with a right parenthesis substituted for the first letter:)ATASET.NAME for DATASET.NAME. The entry is written with today's date as its expiration date.

If QDAS is running, no space holder record is created. Instead, BIM-EPIC reserves the space in the QDAS map.

Close Processing

When a program requests close processing for its datasets, some records can be left in the buffer. These are written to disk when VSE closes the dataset. In some cases there are too many records in the buffer to fit in the space remaining in the current extent, and another extent is necessary to accommodate the excess.

When BIM-EPIC closes a disk dataset, it takes the buffer into account. It determines from the buffer size and from the amount of space left in the extent whether another extent may be necessary. If another extent is needed, BIM-EPIC makes the allocation.

If VSE runs out of room when writing the buffered records, it checks the VSE Label Area for another extent. It will find the extent that BIM-EPIC has placed there. VSE then reads the VTOC, finds an expired dataset there, and rewrites the VTOC entry with the real dataset name.

If the extent isn't necessary, the VTOC entry remains as a space holder record. Since the entry indicates the dataset is expired, the space can be used for any uncontrolled dataset at any time, and BIM-EPIC reuses space holder extents on an as-needed basis.

You may see space holder records in your VTOC listings. You may also see an extra extent in the DSN Catalog entry for the version. Neither of these adversely affects processing.

When closing an output disk dataset:

BIM-EPIC determines whether a precautionary extent is required. If so, it writes the space holder record, updates the VSE Label Area with the new extent information, and writes the extent record in the DSN Catalog.

It places the version in closed status in the DSN Catalog.

BIM-EPIC passes control to VSE.

2

BIM-EPIC Online Manager

This chapter describes the BIM-EPIC Online Manager.

Chapter Contents

Introduction to Manual Cataloging	2-2
Introduction to Online Manager	2-3
Using the Online Manager	2-7
Managing BIM-EPIC Dataset Definitions (EDDs)	2-18
Managing Dataset Versions.....	2-30
Managing Tapes	2-40
Retention and Scratching	2-46

Introduction to Manual Cataloging

Description

There are several reasons for doing manual cataloging and catalog management. Automatic cataloging is global in scope, so it will not match the exact requirements of all datasets. You may want to:

- ensure that precise catalog defaults are established before the dataset is written for the first time
- update a definition after it is created in a job
- change a dataset's requirements due to changes in government regulations or datacenter procedures

For manual cataloging, BIM-EPIC provides a panel-driven system called the Online Manager. Online Manager is easy to use, and its full-screen fill-in-the-blank formats lead you through the catalog management task step-by-step.

BIM-EPIC also provides a command-driven catalog management program called TSIDMNT for batch cataloging of new datasets and versions, for updating fields in existing dataset entries, and for DSN Catalog reporting. TSIDMNT is discussed in Chapter 3, "DSN Catalog Management Utility".

BIM-EPIC programs do not require leading zeros; numeric values are zero-filled on the left. However, in Online Manager and TSIDMNT, you can suppress the zero-fill by entering a volume serial number as a six-character string with blanks on the right (Example: "12 ").

Using Online Manager with BIM-EPIC for CMS

If you are using BIM-EPIC Online Manager under CMS, we recommend defining at least four (4) megabytes of virtual storage for your CMS machine.

Sharing a BIM-EPIC Catalog with CA-EPIC for CMS

If you run both BIM-EPIC and CA-EPIC for CMS, and share a DSN Catalog on VSE, you can update *tape* datasets from your CMS machines. You can also scratch tapes used for CA-EPIC for CMS backup/archive datasets.

Introduction to Online Manager

Overview

Description

This chapter gets you started with online management, including:

- accessing Online Manager
 - using panels and menus
 - getting help
 - exiting Online Manager
-

Online Access Using CICS

Online Manager can be accessed as a CICS transaction. See the *BIM-EPIC Installation and System Guide* for information on installing the CICS EPIC transaction. If online access will be through BIM-FAQS/ASO only, installing the CICS EPIC transaction is not required.

Online Access Using BIM-FAQS/ASO

If BIM's BIM-FAQS/ASO product is installed, and BIM-EPIC is defined to it, the Online Manager can be accessed by entering EPIC from any BIM-FAQS/ASO command line. If online access will be through BIM-FAQS/ASO only, installing the CICS EPIC transaction is not required.

Online Access From CMS

When using CA-EPIC for CMS, the Online Manager can be accessed by entering **EPICCMS** from CMS. This requires a VM user ID with the authority to access Online Manager.

Primary Option Menu

When the Online Manager is accessed, the Primary Option Menu is displayed first.

```
TSIO0001.O ----- BIM-EPIC - Primary Option Menu ----- USER=TR91
OPTION  ===>

      OPTION      COMMAND      DESCRIPTION
      M           MANAGE       Search and manage BIM-EPIC Catalog
      D           DEFINE       Define BIM-EPIC dataset (EDD)
      S           SYSTEM       Manage BIM-EPIC system configuration
      P           PROFILE       Manage BIM-EPIC user profiles
      C           CAPACITY      Display BIM-EPIC Catalog statistics

      H           HELP         Learn about online help
      X           EXIT         Exit BIM-EPIC

      F1=Help  F3=Return  F5=Recall  F12=Exit
```

Using Menus

When the Primary Option Menu is displayed, press PF1 to read the help information for the primary online features. Select the H (Help) option to learn more about online help capabilities.

Using Online Help

Online Manager provides a number of Help options. Choose the H (Help) option on the Primary Option Menu to learn about these. Whether the cursor is located on a panel’s command area or an input field, the PF1 key always provides help information.

Exiting the Online Manager

The X (Exit) command on the Primary Option Menu ends the online session.

Online Catalog Management Commands

Description	Even while using Online Manager panels for your catalog management, you can still execute TSIDMNT functions in command mode. To do this, place the cursor in a panel's command area, and type EPIC , followed by a space and the normal TSIDMNT syntax. See Chapter 3, "DSN Catalog Management Utility", for more information.
DSN Catalog Usage Display	Use CAPACITY to display statistics on current DSN Catalog file use.
Cataloging New Datasets	There are two commands for entering new files in the catalog. Use DEFINE to create new BIM-EPIC dataset definitions (EDDs). Use ADD to enter new versions to cataloged datasets.
Updating Datasets	<p>There are several commands available for modifying the characteristics of existing DSN Catalog entries.</p> <p>UPDATE modifies the BIM-EPIC dataset definition (EDD).</p> <p>RETAIN modifies creation and expiration data for a version.</p> <p>CYCLE modifies retention for a selected version and all previous versions.</p> <p>CLOSE changes open and conditionally cataloged versions to active status.</p> <p>LOCK locks datasets and versions, UNLOCK unlocks locked datasets and versions, and RELEASE releases locked datasets and versions for one use only.</p>
Scratching Datasets	There are three commands for scratching files: SCRATCH, DELETE, and PURGE. DELETE deletes the dataset definition and all dataset versions. PURGE deletes the dataset definition and all its versions, and deletes associated volume serial numbers from the DSN Catalog. SCRATCH deletes one version only.
Changing the Name of a EDD	Change a BIM-EPIC dataset definition name with RENAME. Transfer a version from one dataset name to another with TRANSFER. You can make a tape version accessible with two dataset names using MCATALOG.
Tape Management Commands	CLEAN records tape cleaning.

**Disk Volume
Control (BIM-EPIC
only)**

Using TSIDMNT commands, you can temporarily remove disk packs from BIM-EPIC processing. Use QUIESCE to prevent writing controlled output datasets on a disk volume, while still allowing datasets to be read from it. Use OFFLINE to remove the volume from both input and output processing. ONLINE returns the volume to full processing.

Using the Online Manager

Configuring Online Security

Controlling Access to BIM-EPIC Online Manager

Online Manager has its own panel-driven security system. This system allows online catalog management functions and some BIM-EPIC system functions to be restricted selectively to individual users, and for multiple users to inherit (model) an individual set of security attributes. It also provides a set of security defaults for users who are not defined to BIM-EPIC online security.

Defining a security profile for every potential user at a large site can be a tedious task. BIM-EPIC Online Manager security overcomes this problem by allowing flexible combinations of global defaults, group modeling and unique individual profiles. One site may wish to grant wide system access and restrict only a few users, while another may want widespread restrictions and limited user access. All are easily accomplished with BIM-EPIC online security.

BIM-EPIC Online Manager is supplied unsecured. A default security profile (\$DEFSEC) automatically governs all unsecured users; its global security defaults can be modified as needed. Users with security system access can also use a non-modifiable unsecured profile (\$EPIC) as an initial model for additional profiles. The first time you modify one profile that was modeled on another, the modified profile becomes independent. If you modify any profile on which others are modeled, the changes are inherited by the others. In this way, any combination of independent and related security profiles can be created as desired.

Online Help

Online help is provided for the security panels and input fields. To receive help, place the cursor on the first character of the command area or an input field and press PF1.

Planning Your Online Manager Security

There are some issues you need to consider if you plan to use Online Manager security. First, the online security profile names you create must follow the convention appropriate for your operating environment. For example, if you are running BIM-EPIC under BIM-FAQS/ASO, user IDs defined for BIM-EPIC security must be known to BIM-FAQS/ASO.

Second, you should always reserve exclusive security system access for your security administrator and then secure \$DEFSEC itself against security system access (see below).

Third, you should determine whether the global default settings (\$DEFSEC) will be adequate for most users, with occasional exceptions to be defined, or whether you need a number of different security profiles for specific users or groups of users (by department, for example).

Accessing the Online Security System

To access the security system from the Primary Option Menu, type SECURITY at the command line and press ENTER, or select the SYSTEM option to display the System Configuration Menu. To access the security system from the System Configuration Menu, select the SECURITY option.

Secured Users Panel

The Secured Users panel displays the names of security profiles that are defined to Online Manager. Users authorized to do so can use the Secured Users panel to create, modify, and model security profiles.

```
TSIO0800.T ----- BIM-EPIC - Secured Users ----- USER=TR91
COMMAND ===>

      User ID      (MODEL)      User ID      (MODEL)      User ID      (MODEL)
      _ $DEFSEC          _ $EPIC          _ SJA          $EPIC

L = Delete   M = Model   U = Update
F1=Help  F3=Return  F5=Recall  F6=Add  F12=Exit
```

Default Security Profile (\$DEFSEC)

The \$DEFSEC profile is supplied unsecured and always acts as the default security profile for all unsecured users, so that the security administrator does not have to define a profile for every user. To prevent unauthorized access to Online Manager, \$DEFSEC cannot be used as a user ID even if left unsecured. If Bypass the BIM-EPIC Sign-On Panel is set to 'N' in \$DEFSEC, all unsecured users will be locked out of Online Manager unless they know a secured ID and password.

The \$EPIC Security Profile

\$EPIC is an unmodifiable, unsecured profile that is supplied for use as a model. After \$DEFSEC has been secured, you can grant total system access to selected users by modeling their user IDs after \$EPIC. Users modeled on \$EPIC can themselves be modified to create customized security profiles, as with any other model. To prevent unauthorized access to Online Manager, \$EPIC cannot be used as a user ID. Use caution when modeling user profiles after \$EPIC and leaving them unmodified, because whoever learns one such profile name could use the Sign-On panel to gain total system access!

Creating a New User Security Profile

To create a brand new Secured User, use an existing security profile as the initial model.

Example

Enter SECURITY at an Online Manager command line to display the Secured Users panel. Place M (for Model) in front of the \$DEFSEC profile, and press ENTER. When prompted for a new profile name, enter USER99. The list displays USER99 and shows that it is modeled on \$DEFSEC. USER99 now possess the security attributes of \$DEFSEC.

The User Security Panel

Users authorized to do so can use the User Security panel to modify and model security settings for a selected user or group ID. The panel allows changing the security password, setting user global options for security access, and enabling or disabling specific BIM-EPIC DSN Catalog management functions for a user or group ID.

```
TSIO0801.u ----- BIM-EPIC - User Security ----- USER=TR91
COMMAND ==>

USER INFORMATION                                DSN CATALOG COMMAND ALLOWED?
UserID:      SJA                                Add      ==> Y   Clean    ==> Y
Password ==>                                     Close    ==> Y   Cycle   ==> Y
Verify  ==>                                     Define   ==> Y   Delete  ==> Y
Model    ==> $EPIC                               Display  ==> Y   Lock    ==> Y
                                                MCatalog ==> Y   Move     ==> Y
                                                Purge    ==> Y   Release ==> Y
USER GLOBAL OPTIONS                               Rename   ==> Y   Retain  ==> Y
Security system access ==> Y                     Scratch  ==> Y   Transfer ==> Y
Bypass the SIGN ON panel ==> Y                     Unlock   ==> Y   Update  ==> Y
VSE System option access ==> Y                     Poolcmd: Off/Onl/Qui ==> Y
Profile system access ==> Y
Confirmation changes? ==> Y
Other users' profiles? ==> Y

F1=Help  F3=Return  F5=Recall  F6=Update Rules  F12=Exit
```

User Security Panel Fields

USER INFORMATION - When the BIM-EPIC Sign-On panel has been enabled for a user (see below), a user password will be requested. The Password and Verify fields can be used to create and modify this password. An asterisk (*) at the end of these fields indicates an existing password. For privacy, neither entry is displayed on the screen. The Model field can be used to import security settings from another secured user.

USER GLOBAL OPTIONS - The Security system access field determines whether the selected user can make security system changes. The Bypass the SIGN ON panel field determines whether access to Online Manager is controlled by the BIM-EPIC Sign-On Panel (see example below). In BIM-EPIC only, the VSE System option access field controls access to BIM-EPIC's processing options. The Profile system access field controls access to the user's PROFILE display formats. Subsequent fields can be used to restrict the user's access to their own confirmation settings and other users' profiles, if desired. We recommend that access to Other users' profiles be set to 'N', so that profile modeling cannot be affected by anyone besides the owner of a model profile.

DSN CATALOG COMMANDS - These fields control access to individual DSN Catalog management functions.

Changing Security Settings for One User

Enter SECURITY at an Online Manager command line to display the Secured Users panel. Place U (for Update) in front of the selected user's security profile, and press ENTER. When the User Security panel is displayed, make the necessary changes. Press PF6 to accept the changes. If the security profile was formerly modeled after another, this relationship no longer exists, as reflected in the now-empty Model fields for this user.

Sign-On Panel

The Sign-On panel is used to require a password from a secured user. The panel prompts the user for a secured user ID and the appropriate password. The panel will only recognize user IDs listed on the Secured Users panel. If Bypass the SIGN On panel is set to 'N' in \$DEFSEC, all unsecured users are locked out of Online Manager unless they know a secured user ID and password.

```
TSIOSIGN.0 head                                USER=SJA
COMMAND ==>>

      BBBB      IIIII      M      M      OOO      Y      Y      L      EEEEE
      B  B      I      MM      MM      O      O      Y      Y      L      E
      B  B      I      M M M M      O      O      Y Y      L      E
      BBBB      I      M M M M      O      O      Y      L      EEEE
      B  B      I      M      M      O      O      Y      L      E
      B  B      I      M      M      O      O      Y      L      E
      BBBB      IIIII      M      M      OOO      Y      LLLLL EEEEE

      BIM-EPIC - ONLINE MANAGER

      User ID ==>> _____
      Password ==>> _____

      F1=Help  F3=Return  F5=Recall  F12=Exit
```

**Security
Administration
Example**

Step	Action
1	Use your security administrator's user ID as the name for a new security profile modeled on \$EPIC.
2	Update the new administrator profile with password protection and enable password prompting by setting Bypass the BIM-EPIC Sign-On Panel to 'N'. Be sure to update rules with PF6 before exiting from the modified administrator profile (if you do not do this, your administrator's profile may still model \$EPIC).
3	Modify \$DEFSEC to deny all system access (including security) and catalog update functions to unsecured users.
4	Create any desired group security profiles. For instance, you can create a departmental profile named DEPT1 and provide it with desired catalog update access for that department. Then model several key departmental user IDs after DEPT1.
5	Create any additional customized user security profiles as desired.

Maintaining User Profiles

Description

Online Manager allows users to customize the way dataset definition and version information is displayed. BIM supplies two default profiles containing display layouts and confirmation options. The \$DEFPROF default profile provides a modifiable layout that applies to all users who are not specifically defined to Online Manager. The \$EPIC default profile provides a fixed standard layout. Users can model either default profile under a name of their choice and modify the characteristics of the new profile as desired. Whenever you model an existing user profile, the new profile's attributes can then be modified, but this "disconnects" it from inheriting the model's attributes in the future.

Access

From the Primary Option Menu, choose P for PROFILE. You can also type PROFILE at the command line on most panels and press ENTER.

User Profiles Panel

```
TSIO0300.0 ----- BIM-EPIC - User Profiles ----- USER=SJA
COMMAND ==>
The SJA profile has been added using the $EPIC profile as a model.

      Profile      (MODEL)      Profile      (MODEL)      Profile      (MODEL)
      _ $DEFPROF      _ $EPIC      _ SJA      $EPIC

Active Profile:  $DEFPROF

A = Activate   L = Delete   M = Model   U = Update
F1=Help   F3=Return   F5=Recall   F6=Add   F12=Exit
```

From this panel, you can create and modify user profiles with different panel display formats and confirmation options. Press PF1 to display help information on using User Profiles panels.

Select M (for Model) to create a copy of the \$EPIC or any other existing profile, and give the copy another name.

Use U (for Update) to modify the attributes of an existing profile. The first of three attribute panels is displayed. After you have changed any of your profile defaults, press ENTER to update your changes. Press PF3 to exit without changes.

Customizing the EDD Display

Description

Use the Modify EDD Display panel to change the screen display format for BIM-EPIC dataset definitions.

The Current Display Order area provides functions to Move (M) a field to a new location Following (F) or Preceding (P) another field, or to Remove (R) a field from the display. Press PF1 for more information.

```
TSIO0301.u ----- BIM-EPIC - User Profile Definition ----- USER=SJA
COMMAND ==>
The SJA profile is modeled after $EPIC.
                ** MODIFY EDD DISPLAY **

CURRENT DISPLAY FORMAT                                     More: +>

Dataset Name          Cyc Reten  DLA ----- Attributes ----- GEN or
Width ==> 25                               Lock Rel Rev Unlbl NSub  WRK

CURRENT DISPLAY ORDER  Profile: SJA          Model: $EPIC

- 1 Cyc                - 2 Reten              - 3 DLA                - 4 Attributes
- 5 GEN or WRK         - 6 Tape Pool           - 7 Tape Mode         - 8 Vault Method
- 9 Prime Dataset     - 10 Owner ID            - 11 AGP              - 12 Alloc Req
- 13 SYSnnn           - 14 Reblock Value       - 15 Comment          - 16 User ID
- 17 System ID        - 18 Last Output         - 19 Create Date      - Disk Pool
- Primary Alloc       - Sec Alloc              - Max Sec

F = Following  M = Move  P = Preceding  R = Remove
F1=Help  F3=Return  F6=Update  F8=Fwd  F11=Right  F12=Exit
```

Press F6 to accept the new display order. The Current Display Format area will change to show the new arrangement. Press F8 to scroll to additional User Profile Definition panels.

Customizing the Version Display

Description

Use the Modify Version Display panel to change the screen display format for dataset version information.

The Current Display Order area provides functions to Move (M) a field to a new location Following (F) or Preceding (P) another field, or to Remove (R) a field from the display. Press PF1 for more information.

```
TSIO0302.8 ----- BIM-EPIC - User Profile Definition ----- USER=SJA
COMMAND ==>

                                ** MODIFY VERSION DISPLAY **

CURRENT DISPLAY FORMAT                                         More: -+>

Dataset Name          Version Number-of First  --- Creation ---
Width ==> 25          Vols/Exts Volser Date      Time

CURRENT DISPLAY ORDER    Profile: SJA          Model: $EPIC

- 1 Version             - 2 Exts/Vols          - 3 First Volser    - 4 Create Date
- 5 Create Job          - 6 Expire Date         - 7 Gen Number     - 8 Status
- 9 Last Acc Info      - 10 Media Type         - 11 Attributes    - 12 Extent Info
- 13 File Seq          - 14 Vault Method       - 15 Slot          - 16 Blocks
- 17 Block Size        - 18 Record Size       - 19 Tape Pool     - 20 Tape Mode
- 21 GEN or WRK        - 22 SYSLOG ID         - 23 CPU ID        - 24 Reel
- Create Info          - Last Acc Date      - Last Acc Job     - Reten Info
- Reten                - Cyc              - DLA              - Last Backup

F = Following  M = Move  P = Preceding  R = Remove
F1=Help  F3=Return  F6=Update  F7=Bkwd  F8=Fwd  F11=Right  F12=Exit
```

Press F6 to accept the new display order. The Current Display Format area will change to show the new arrangement. Press F8 to scroll to the Modify Profile Defaults panel. Press F7 to scroll back to the Modify EDD Display panel.

Customizing the Confirmation Options

Description

Use the Modify Profile Defaults panel to enable or disable confirmation prompts to a variety of BIM-EPIC functions, and to change the default level for a catalog search. The \$EPIC default profile always requests confirmation.

```
TSIO0303.8 ----- BIM-EPIC - User Profile Definition ----- USER=SJA
COMMAND ==>

                ** MODIFY PROFILE DEFAULTS **
                More: -

Profile: SJA      Model: $EPIC

      CONFIRMATION SETTINGS                LEVEL FOR CATALOG SEARCH

Add Ver  ==> Y  Clean   ==> Y  Level ==> E  E = EDD
Close   ==> Y  Cycle   ==> Y                V = Version
Define  ==> Y  Delete  ==> Y                ( X = Exploded list )
Lock    ==> Y  MCatalog ==> Y
Move    ==> Y  Purge   ==> Y
Release ==> Y  Rename  ==> Y
Retain  ==> Y  Scratch ==> Y                SYSTEM PASSWORD PROMPT
Sys Opts ==> Y  Transfer ==> Y                Prompt on initial entry ==> Y
Unlock  ==> Y  Update  ==> Y

F1=Help  F3=Return  F6=Update  F7=Bkwd  F12=Exit
```

Change confirmation settings and Search Catalog level as desired. Press F6 to accept the new settings. Press F7 to scroll back to previous User Profile Definition panels.

Using Online Manager Help

Description	<p>The Online Manager provides a number of types of online help for the user. These include:</p> <ul style="list-style-type: none">• Panel help• Input field help• Command help• Panel navigation help
Help for Help	<p>The H (Help) option of the Primary Option Menu provides a description of the Help system.</p>
Panel Help	<p>Press PF1 in the command line of any panel to display help for that panel.</p>
Field Help	<p>Place the cursor at the beginning of any input field and press PF1 to display online help for that field.</p>
Command Help	<p>The Online Manager provides help in understanding BIM-EPIC management commands. To view a list of these commands, enter CMDHELP at the command line on any panel.</p>
Panel Navigation Help	<p>The Online Manager provides help in understanding panel navigation. To view a diagram of its panel system, enter the command EPICFLOW at the command line on any panel. The Online Flow panel below is displayed along with a highlighted path showing how the user got to their current location. Help for any panel on this diagram can be viewed by entering its panel ID on the command line in the Online Flow panel and pressing ENTER.</p>

Panel IDs Display (I)

```

TSIO0901.b ----- BIM-EPIC - Online Flow ----- USER=SJA
COMMAND ==>

                                TSIO0001
                                v
                                EPICFLOW Titles ==> I (D/I)
+-----+-----+-----+-----+-----+-----+
| TSIO0300 | | TSIO0500 | | TSIO0100 | | TSIO0400 | |
| v | | | | | v | | | | | v | |
+-----+-----+-----+-----+-----+-----+
| TSIO0103 | <-+ | | | | | | | | | | TSIO0401 | <-+ |
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| TSIO0301 | TSIO0104 | <-+ | +--< TSIO0101 | >--+ | | | | | TSIO0205 |
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| TSIO0302 | TSIO0105 | <-+ | +--< TSIO0102 | <--+ | | | | | TSIO0206 |
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| TSIO0303 | TSIO0106 | <-+<--+ | | | | | | | | | | TSIO0207 | <--+ |
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| TSIO0202 | <-+ | | | | | | | | | | TSIO0200 | <--+ |
| v | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| TSIO0203 | <-+ | | | | | | | | | | TSIO0201 | <--+ |
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| TSIO0204 | <-+ | | | | | | | | | | TSIO0212 | <--+ |
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| TSIO0204 | <-+ | | | | | | | | | | TSIO0213 | <--+ |
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| TSIO0211 |
+-----+-----+-----+-----+-----+-----+
F1=Help F3=Return F5=Recall F11=Right F12=Exit

```

To view the Online Flow in terms of its panel functions, enter D (for Descriptions) on the EPICFLOW Titles field and press ENTER to display the panel below:

Panel Descriptions Display (D)

```

TSIO0901.b ----- BIM-EPIC - Online Flow ----- USER=SJA  COMMAND
====>

                                MAINMENU
                                v
                                EPICFLOW Titles ==> D (D/I)
+-----+-----+-----+-----+-----+-----+
| PROFILE | | CAPACITY | | MANAGE | | TAPE MGT | |
| v | | | | | v | | | | | v | |
+-----+-----+-----+-----+-----+-----+
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| E FORMAT | UPDATE | <-+ | +--< EDD MGMT | >--+ | | | | | VOL INFO | <-+ |
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| V FORMAT | RENAME | <-+ | +--< EDD INFO | <--+ | | | | | MCATALOG |
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| CONFIRM | PASSWORD | <-+<--+ | | | | | | | | | | TRANSFER |
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| ADD VER | <-+ | | | | | | | | | | VER MGMT | <--+ |
| v | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| ADDTAPEV | <-+ | | | | | | | | | | VER INFO | <--+ |
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| ADDDISKV | <-+ | | | | | | | | | | APP TAPE | <--+ |
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| ADDDISKV | <-+ | | | | | | | | | | APP EXT | <--+ |
| | | | | | | | | | | | | | | | | | | | |
+-----+-----+-----+-----+-----+-----+
| RETAIN |
+-----+-----+-----+-----+-----+-----+
CYCLE
F1=Help F3=Return F5=Recall F11=Right F12=Exit

```

Managing BIM-EPIC Dataset Definitions (EDDs)

Description

BIM-EPIC maintains a dataset-level description for each unique dataset name it controls, regardless of the number of versions of it that exist. This BIM-EPIC dataset definition (EDD) can be created automatically during processing, or it can be defined manually.

Displaying BIM-EPIC Dataset Definitions

Introduction

Use the Search Catalog panel to search for dataset definitions in the DSN Catalog. You can search for EDDs using the following criteria:

- Two-character user-defined User ID (not your system USERID)
- Two-character user-defined System ID (not the SMFID)
- Tape pool ID

Using the Manage EDDs panel, you can view, modify, and update dataset definitions.

Access

From the Primary Option Menu, select M (for MANAGE) and press ENTER to display the Search Catalog panel.

Search Catalog Panel

```

TSIO0100.M ----- BIM-EPIC - Search Catalog ----- USER=SJA
COMMAND ==>
Specify a dataset/token or volser.

CATALOG SEARCH
Name ==> _____
Level ==> E          E = EDD
or
Volser ==> _____ V = Version ( X = Exploded List )

DISPLAY FORMAT
Active Profile ==> $DEFPROF (1-8 characters)
Profile Level ==> _          E = EDD
                              V = Version
                              C = Confirmation

EDD FILTERS                      VERSION FILTERS
User ID ==> _ (2 chars)          Tape or Disk ==> _ (T/D)
System ID ==> _ (2 chars)        Scratched ==> _ (Y/N)
Tape Pool ==> _ (0-9 or A-Z)     Create Job ==> _____

F1=Help F3=Return F5=Recall F12=Exit

```

Procedure

Step	Action
1	Type a fully or partially qualified dataset name in the Name field. If you specify a partial dataset name, all EDDs that meet the name criteria are displayed. An asterisk (*) can be substituted for dataset levels. A question mark (?) can be substituted for any individual character.
2	Enter an 'E' in the Level field to view selected EDDs.
3	If you want more specific search criteria, use the EDD filters at the bottom of the panel.
4	Press ENTER to display the Manage EDDs panel listing the dataset definitions that match your search criteria. Or press PF3 to exit the Search Catalog panel without listing dataset definitions.

Managing EDDs

Introduction

The Manage EDDs panel lists dataset definitions matching the dataset name or expression you entered in the Search Catalog panel. From this panel, you can select a dataset definition to display or update. You can add, delete, lock, purge, release or unlock a dataset definition. You can also request a list of its versions.

Access

From the Primary Option Menu, press M (for MANAGE). Complete the Search Catalog panel and press ENTER to display the Manage EDDs panel.

Manage EDDs Panel

```
TSIO0101.Z ----- BIM-EPIC - Manage EDDs ----- USER=SJA
COMMAND ===>

                                     More: >

Dataset Name          Cyc Reten  DLA ----- Attributes ----- GEN or
Width: 25              Lock Rel  Rev Unlbl NSub  WRK

- BATCH CONVERT OUT      1      0      0 N      N      N      N      N      GEN
- BILL BACKUP             4      0      0 N      N      N      N      N      GEN
- BILL.REPRINT.FILE       2      0      0 N      N      N      N      N      GEN
- BILLING.PAN.ORDER       1      0      0 N      N      N      N      N      GEN
- BILRECS BKUP TAPE      60      0      0 N      N      N      N      N      GEN
- BLS.REC                 6      0      0 N      N      N      N      N      GEN
- BUD.ACASH.BKUP          1      0      0 N      N      N      N      N      GEN
- BUDGET.SQLBKUP          3      0      0 N      N      N      N      N      GEN
- BUDGET.SQLBKUP1         3      0      0 N      N      N      N      Y      GEN

A = Add-Ver C = Close F = Release I = Info J = Lock K = Unlock L = Delete
M = Model P = Purge R = Rename U = Update V = Versions ( X = Exploded Lst)
F1=Help F3=Return F5=Recall F11=Right F12=Exit
```

Procedure

Step	Action
1	To manage a dataset definition entry, type the letter of the command you want to execute in front of the dataset name to be processed. You can enter commands for as many entries as you wish.
2	Press ENTER to invoke the commands entered. Press PF3 to exit without performing any commands.

Using TSIDMNT DISPLAY

BIM-EPIC also provides the DISPLAY command of the TSIDMNT catalog management utility to list dataset definitions. See Chapter 3, “DSN Catalog Management Utility”, for information on using the DISPLAY command.

Displaying a Dataset Definition

Introduction The EDD Information panel displays the dataset definition attributes. You can update those attributes or create a new definition based on the one displayed.

Access From the Manage EDDs panel, place an I (Information) in front of a dataset definition entry and press ENTER to display the EDD Information panel.

EDD Information Panel

```
TSIO0102.G ----- BIM-EPIC - EDD Information ----- USER=SJA
COMMAND ===>

( _ ) Dataset: BATCH CONVERT OUT

      TAPE INFORMATION          DATASET INFORMATION
TPAdev: N  Mode: NO            GEN or WRK: GEN  BLKSZ:      OPT
Tape Pool: .                  Reten:      0    Create Date: 01/31/1997
Vault Method: 0               Cycles:     1    Last Output: 08/01/1997
Owner ID: .                   DLA:        0    System ID:  .
Tape Size:  LARGE             SYSnnn:     0    User ID:   .

      DISK INFORMATION          DATASET ATTRIBUTES
Disk Pool:  DISK02            Prime:       Y    Revolving:   N
Primary:    50                Locked:     N    Unlabeled:   N
Secondary:  25                Newsub:    N    Released:   N
Max Sec:    255               Pswd Prot: N
Primary ALR: -                Comment:    .
Auto Purge: N                Prime DSN:  .

A = Add-Ver C = Close F = Release J = Lock   K = Unlock   L = Delete
M = Model   P = Purge R = Rename  U = Update V = Versions ( X = Exploded Lst)
F1=Help    F3=Return  F5=Recall  F12=Exit
```

Step	Action
1	If you want to manage the dataset definition displayed, type the desired command in front of the dataset name.
2	Press ENTER to execute the desired function. Press PF3 to exit without performing any commands.

Using TSIDMNT DISPLAY BIM-EPIC also provides the DISPLAY command of the TSIDMNT catalog management utility to list dataset definitions. See Chapter 3, “DSN Catalog Management Utility”, for information on using the DISPLAY command.

Creating a Dataset Definition

Introduction

Use the Define an EDD panel to create new dataset definitions. The attributes on this panel are those of the dataset being modeled if you typed M in the input field of the EDD Information panel.

Access

You can access the Define an EDD panel by entering D (for DEFINE) on the Primary Option Menu. Or enter M (for Model) in front of the dataset definition name on either the Manage EDDs or EDD Information panels.

Define an EDD Panel

```
TSIO0103.D ----- BIM-EPIC - Define an EDD ----- USER=SJA
COMMAND ==>

DEFINE

Dataset ==> _____
Model   ==> _____

TAPE INFORMATION      DISK INFORMATION      DATASET INFORMATION
TPAdev: N  Mode:  _   Disk Pool   ==>  _   GEN or WRK   ==>  _
Tape Pool  ==>  _   Primary     ==>  _   Retention   ==>  _
Vault Meth ==>  _   Secondary   ==>  _   Cycles      ==>  _
Owner ID   ==>  _   Max Sec     ==>  _   DLA         ==>  _
Unlabeled  ==>  _   Primary ALR ==>  _   BLKSZ( _ CISZ) ==>  _
Tape Size  ==>  _   Auto Purge  ==>  _   SYSnnn     ==>  _
Revolving  ==>  _                               System ID  ==>  _
                                           User ID     ==>  _
Comment    ==>  _   Password    ==>  _
Prime DSN  ==>  _   Encrypt Key ==>  _

F1=Help  F3=Return  F5=Recall  F12=Exit
```

Procedure

Step	Action
1	To create a new EDD from scratch, specify a dataset name in the Dataset field. The dataset name is the key BIM-EPIC uses to locate a record in the DSN Catalog. For this reason, the dataset name must be unique. It must also follow all CMS naming rules. Use the Model field when you are defining a new dataset which is similar to a previously defined dataset.
2	Update any desired attributes for the dataset. Any modifications you make on the Create an EDD panel override modeled attributes.
3	Press ENTER to create the new EDD. All data entered are verified for syntactic correctness and defaults added. Press PF3 to exit the panel without creating the EDD.

**Using TSIDMNT
CATALOG**

BIM-EPIC also provides the CATALOG command of the TSIDMNT catalog management utility to create a BIM-EPIC dataset definition (EDD). See Chapter 3, “DSN Catalog Management Utility”, for information on using the CATALOG command.

Updating a Dataset Definition

Introduction

Online Manager provides the Update an EDD panel to update your BIM-EPIC dataset definitions (EDDs) in the DSN Catalog. The attributes displayed on this panel are the defaults for all output datasets using this dataset name. Any modifications affect subsequent versions only.

Access

Select M (for MANAGE) from the Primary Option Menu, and use the Search Catalog panel to specify dataset name criteria. From the Manage EDDs list displayed, place a U next to the name of the dataset definition you wish to update and press ENTER.

Update an EDD Panel

TSIO0104.e ----- BIM-EPIC - Update an EDD ----- USER=SJA

COMMAND ==>

UPDATE

Dataset: BATCH CONVERT OUT

TAPE INFORMATION		DISK INFORMATION		DATASET INFORMATION	
TPAdev: N	Mode: NO	Disk Pool	==> DISK02	GEN or WRK	==> GEN
Tape Pool	==> _____	Primary	==> 50	Retention	==> 0
Vault Meth	==> 0	Secondary	==> 25	Cycles	==> 1
Owner ID	==> _____	Max Sec	==> 255	DLA	==> 0
Unlabeled	==> N	Primary ALR	==> _____	BLKSZ(_ CISZ)	==> OPT
Tape Size	==> L	Auto Purge	==> N	SYSnnn	==> 0
Revolving	==> N			System ID	==> _____
				User ID	==> _____
Comment	==> _____			Password	==> _____

F1=Help F3=Return F5=Recall F12=Exit

Procedure

Step	Action
1	Update any desired attributes for the dataset definition. To remove the Disk Pool specification, replace the current value with the word "NO". To remove the "Primary", "Secondary", "Max Sec", or "Primary ALR" value, replace the current value with a zero (0).
2	Press ENTER to update the EDD, or press PF3 to exit without updating.

Using TSIDMNT UPDATE

BIM-EPIC also provides the UPDATE command of the TSIDMNT catalog management utility to modify a BIM-EPIC dataset definition. See Chapter 3, “DSN Catalog Management Utility”, for information on using the UPDATE command.

Locking a Dataset Definition

Introduction Locking a BIM-EPIC dataset definition prevents its use for output (creating a new version with the same dataset name). You can lock a dataset definition from the Manage EDDs panel.

Access From the Primary Option Menu, press M (for MANAGE). Complete the Search Catalog panel and press Enter.

Manage EDDs Panel

```
TSIO0101.Z ----- BIM-EPIC - Manage EDDs ----- USER=SJA
COMMAND ===>

More: >

Dataset Name          Cyc Reten  DLA ----- Attributes ----- GEN or
Width: 25              Lock Rel Rev Unlbl NSub  WRK
- BATCH CONVERT OUT    1      0    0 N      N      N      N      N      GEN
- BILL BACKUP           4      0    0 N      N      N      N      N      GEN
- BILL.REPRINT.FILE     2      0    0 N      N      N      N      N      GEN
- BILLING.PAN.ORDER     1      0    0 N      N      N      N      N      GEN
- BILRECS BKUP TAPE    60      0    0 N      N      N      N      N      GEN
- BLS.REC               6      0    0 N      N      N      N      N      GEN
- BUD.ACASH.BKUP        1      0    0 N      N      N      N      N      GEN
- BUDGET.SQLBKUP        3      0    0 N      N      N      N      N      GEN
- BUDGET.SQLBKUP1       3      0    0 N      N      N      N      Y      GEN

A = Add-Ver C = Close F = Release I = Info  J = Lock  K = Unlock L = Delete
M = Model   P = Purge R = Rename  U = Update V = Versions ( X = Exploded Lst)
F1=Help    F3=Return  F5=Recall  F11=Right F12=Exit
```

Step	Action
1	Place the cursor in front of the name of the dataset definition you wish to lock.
2	Type J (Lock) on the line and press Enter.

Using TSIDMNT LOCK BIM-EPIC also provides the LOCK command of the TSIDMNT catalog management utility to lock a BIM-EPIC dataset definition. See Chapter 3, “DSN Catalog Management Utility”, for information on using the LOCK command.

Releasing a Locked Dataset Definition

Introduction

Releasing a locked BIM-EPIC dataset definition permits one use of the EDD for output (creating one new version with the EDD name and attributes). You can release a dataset definition from the Manage EDDs panel. To permanently release an EDD, use the UNLOCK procedure.

Access

From the Primary Option Menu, press M (for MANAGE). Complete the resulting Search Catalog panel and press Enter.

Manage EDDs Panel

```
TSIO0101.Z ----- BIM-EPIC - Manage EDDs ----- USER=SJA
COMMAND ===>

                                     More: >

Dataset Name          Cyc Reten  DLA ----- Attributes ----- GEN or
Width: 25              Lock Rel Rev Unlbl NSub  WRK

- BATCH CONVERT OUT      1      0      0  N      N      N      N      N      GEN
- BILL BACKUP             4      0      0  N      N      N      N      N      GEN
- BILL.REPRINT.FILE       2      0      0  N      N      N      N      N      GEN
- BILLING.PAN.ORDER       1      0      0  N      N      N      N      N      GEN
- BILRECS BKUP TAPE      60      0      0  N      N      N      N      N      GEN
- BLS.REC                 6      0      0  N      N      N      N      N      GEN
- BUD.ACASH.BKUP          1      0      0  N      N      N      N      N      GEN
- BUDGET.SQLBKUP          3      0      0  N      N      N      N      N      GEN
- BUDGET.SQLBKU1         3      0      0  N      N      N      N      Y      GEN

A = Add-Ver C = Close F = Release I = Info J = Lock K = Unlock L = Delete
M = Model P = Purge R = Rename U = Update V = Versions ( X = Exploded Lst)
F1=Help F3=Return F5=Recall F11=Right F12=Exit
```

Procedure

Step	Action
1	Place the cursor in front of the name of the the dataset definition you wish to release.
2	Type F (Release) on the line and press ENTER.

Using TSIDMNT RELEASE

BIM-EPIC also provides the RELEASE command of the TSIDMNT catalog management utility to allow one use of a locked BIM-EPIC dataset definition. See Chapter 3, “DSN Catalog Management Utility”, for information on using the RELEASE command.

Unlocking a Dataset Definition

Introduction Unlocking a BIM-EPIC dataset definition allows its unrestricted use for output (creating new versions with the EDD name and attributes). You can lock a dataset definition from the Manage EDDs panel.

Access From the Primary Option Menu, press M for MANAGE. Complete the Search Catalog panel and press ENTER.

Manage EDDs Panel

```
TSIO0101.Z ----- BIM-EPIC - Manage EDDs ----- USER=SJA
COMMAND ==>
More: >

Dataset Name      Cyc Reten  DLA ----- Attributes ----- GEN or
Width: 25              Lock Rel Rev Unlbl NSub  WRK
- BATCH CONVERT OUT      1      0      0 N      N      N      N      N      GEN
- BILL BACKUP             4      0      0 N      N      N      N      N      GEN
- BILL.REPRINT.FILE       2      0      0 N      N      N      N      N      GEN
- BILLING.PAN.ORDER       1      0      0 N      N      N      N      N      GEN
- BILRECS BKUP TAPE      60      0      0 N      N      N      N      N      GEN
- BLS.REC                 6      0      0 N      N      N      N      N      GEN
- BUD.ACASH.BKUP          1      0      0 N      N      N      N      N      GEN
- BUDGET.SQLBKUP          3      0      0 N      N      N      N      N      GEN
- BUDGET.SQLBKUP1         3      0      0 N      N      N      N      Y      GEN

A = Add-Ver C = Close F = Release I = Info  J = Lock  K = Unlock  L = Delete
M = Model   P = Purge R = Rename  U = Update V = Versions ( X = Exploded Lst)
F1=Help    F3=Return  F5=Recall  F11=Right F12=Exit
```

Step	Action
1	Place the cursor in front of the name of the dataset definition you wish to unlock.
2	Type K (Unlock) on the line and press ENTER.

Using TSIDMNT UNLOCK BIM-EPIC also provides the UNLOCK command of the TSIDMNT catalog management utility to unlock a BIM-EPIC dataset definition for unrestricted access. See Chapter 3, “DSN Catalog Management Utility”, for information on using the UNLOCK command.

Deleting a Dataset Definition

Introduction

When you delete a BIM-EPIC dataset definition, all version information is removed from the DSN Catalog, but tape volume information is kept under the dataset name

**** DSN DELETED **** until the volume serial number is reused or purged. You can remove an EDD from the DSN Catalog using the Manage EDDs panel.

Access

From the Primary Option Menu, press M (for MANAGE). Complete the Search Catalog panel and press ENTER.

Manage EDDs Panel

```
TSIO0101.Z ----- BIM-EPIC - Manage EDDs ----- USER=SJA
COMMAND ==>

More: >

Dataset Name          Cyc Reten  DLA ----- Attributes ----- GEN or
Width: 25              Lock Rel Rev Unbl1 NSub  WRK

- BATCH CONVERT OUT      1      0      0 N      N      N      N      N      GEN
- BILL BACKUP            4      0      0 N      N      N      N      N      GEN
- BILL.REPRINT.FILE      2      0      0 N      N      N      N      N      GEN
- BILLING.PAN.ORDER      1      0      0 N      N      N      N      N      GEN
- BILRECS BKUP TAPE     60      0      0 N      N      N      N      N      GEN
- BLS.REC                6      0      0 N      N      N      N      N      GEN
- BUD.ACASH.BKUP         1      0      0 N      N      N      N      N      GEN
- BUDGET.SQLBKUP         3      0      0 N      N      N      N      N      GEN
- BUDGET.SQLBKUP1        3      0      0 N      N      N      N      Y      GEN

A = Add-Ver C = Close F = Release I = Info J = Lock K = Unlock L = Delete
M = Model P = Purge R = Rename U = Update V = Versions ( X = Exploded Lst)
F1=Help F3=Return F5=Recall F11=Right F12=Exit
```

Procedure

Step	Action
1	Place the cursor next to the dataset definition you wish to delete.
2	Type L (Delete) on the line and press ENTER.

Using TSIDMNT DELETE

BIM-EPIC also provides the **DELETE** command of the **TSIDMNT** catalog management utility to remove a BIM-EPIC dataset definition. See Chapter 3, “DSN Catalog Management Utility”, for information on using the **DELETE** command.

Managing Dataset Versions

Displaying Version Information

Introduction

In order to view or maintain a dataset version online, you must first find its entry in the DSN Catalog. Online Manager provides the Search Catalog panel to request a list of dataset versions according to specified criteria. You can search for versions using the following criteria:

- Two-character user-defined User ID (not system USERID)
- Two-character user-defined System ID (not SMFID)
- Whether scratch or active
- Tape pool ID
- Creation job name

Access

From the Primary Option Menu, press M (for MANAGE) to display the Search Catalog panel.

Search Catalog Panel

TSIO0100.3 ----- BIM-EPIC - Search Catalog ----- USER=SJA

COMMAND ==>

CATALOG SEARCH

Name ==> _____

Level ==> E E = EDD

or V = Version (X = Exploded List)

Volser ==> _____

DISPLAY FORMAT

Active Profile ==> \$DEFPROF (1-8 characters)

Profile Level ==> _ E = EDD

V = Version

C = Confirmation

EDD FILTERS

User ID ==> _ (2 chars)

System ID ==> _ (2 chars)

Tape Pool ==> _ (0-9 or A-Z)

VERSION FILTERS

Tape or Disk ==> _ (T/D)

Scratched ==> _ (Y/N)

Create Job ==> _____

F1=Help F3=Return F5=Recall F12=Exit

Procedure

Step	Action
1	Type a fully or partially qualified dataset name in the Name field. If you specify a partial dataset name, all versions that meet the name criteria are displayed. An asterisk (*) can be substituted for dataset levels. A question mark (?) can be substituted for any individual character.
2	Enter an V in the Level field to view selected versions.
3	If you want more specific search criteria, use the version filters at the bottom of the panel.
4	Press ENTER to display the Manage Versions panel listing the dataset versions that match your search criteria. Or press PF3 to exit the Search Catalog panel without listing dataset versions.

**Using TSIDMNT
DISPLAY**

BIM-EPIC also provides the DISPLAY command of the TSIDMNT catalog management utility to list version information. See Chapter 3, “DSN Catalog Management Utility”, for information on using the DISPLAY command.

Selecting a Version from a List

Introduction The Manage Versions panel lists dataset versions based on the search criteria specified on the Search Catalog panel. From Manage Versions, you can select a specific version for display. You can also close, lock, unlock, release, scratch, retain or cycle the version. From Manage Versions, you can also display and maintain tape volume information.

Access Once you have used the Search Catalog panel to enter your search criteria, the Manage Versions panel is displayed, listing all dataset versions that meet the criteria specified. The listing begins with the dataset name you entered. If you did not enter a dataset name, the listing begins with the first dataset in the DSN Catalog meeting any criteria you specified. The size of the DSN Catalog determines how long it takes to retrieve the data.

Manage Versions Panel

TSIO0200.D ----- BIM-EPIC - Manage Versions -----							USER=SJA
COMMAND ===>							
							More: +>
Dataset Name	Version	Number-of	First	---	Creation	---	
Width: 25		Vols/Exts	Volser	Date		Time	
- BATCH CONVERT OUT	1	1	DUMMY6	08/01/1997	18:20		
- BILL BACKUP	1	1	DUMMY1	08/01/1997	19:51		
- BILL BACKUP	2	1	DUMMY1	07/31/1997	20:57		
- BILL BACKUP	3	1	DUMMY1	07/30/1997	19:23		
- BILL BACKUP	4	1	DUMMY3	07/29/1997	19:30		
- BILL.REPRINT.FILE	1	1	DUMMY3	08/01/1997	11:27		
- BILL.REPRINT.FILE	2	1	DUMMY6	07/31/1997	15:05		
- BILLING.PAN.ORDER	1	1	DUMMY0	08/01/1997	17:55		
- BILRECS BKUP TAPE	1	1	000050	07/29/1997	14:14		
- BILRECS BKUP TAPE	2	1	000024	06/26/1997	14:20		
- BILRECS BKUP TAPE	3	1	000061	05/28/1997	16:42		
- BILRECS BKUP TAPE	4	1	000019	04/29/1997	09:57		
- BILRECS BKUP TAPE	5	1	000143	03/28/1997	09:00		
A = Append C = Close E = Ext/Reels F = Release H = Retain I = Information							
J = Lock K = Unlock M = Mcatalog S = Scratch T = Transfer Y = Cycle							
F1=Help F3=Return F5=Recall F8=Fwd F11=Right F12=Exit							

Step	Action
1	To manage a version entry, type the letter of the command you want to execute in front of the version to be processed. You can enter commands for as many entries as you wish.
2	Press ENTER to invoke the commands entered. Press PF3 to exit without performing any commands.

TSIDMNT BIM-EPIC provides the TSIDMNT catalog management utility to manage dataset versions. See Chapter 3, “DSN Catalog Management Utility”, for information on TSIDMNT.

Displaying Version Attributes

Introduction

The Version Information panel lists information on a specific dataset version. From this panel you can add a new version or display information on associated volumes.

Access

From the Manage Versions panel, place an I (Information) next to the dataset name and press ENTER.

Version Information Panel

```
TSIO0201.G ----- BIM-EPIC - Version Information ----- USER=SJA
COMMAND ==>

( _ ) Dataset: BILL.REPRINT.FILE                               Version: 1

VERSION HISTORY
Create Date: 08/01/1997   Expire Date: 08/01/1997   STATUS
Create Time: 11:27       DLA: 0                   Scratched: N
Create Job:  OCNRPRT     Cycles: 2                 Open: N
Last Acc Date: 08/02/1997 Reten: 0                 Locked: N
Last Acc Job: TSIBACK    Last Backup: 000235       Released: N
                                           CondCat: N
                                           Offloaded: N

VERSION INFORMATION
GEN or WRK:  GEN         Block Size: 999           ATTRIBUTES
Generation:  138         Record Size: 3            Newsup: N
File Seq:    -           Blocks: 0                 Revolving: N
SYSLOG ID:   C2          Tape Mode: -              MCataloged: N
CPU ID:      9           Media Type: DISK          Unlabeled: N
Vault ID:    -           Slot: -                   Cycled: N
Tape Pool:   NA          Extents: 1                 Prime: Y

A = Append  C = Close    E = Extents  F = Release  H = Retain  J = Lock
K = Unlock  M = Mcatalog S = Scratch  T = Transfer Y = Cycle
F1=Help    F3=Return    F5=Recall  F12=Exit
```

Using TSIDMNT DISPLAY

BIM-EPIC also provides the DISPLAY command of the TSIDMNT catalog management utility to list version information in batch mode. See Chapter 3, “DSN Catalog Management Utility”, for information on using the DISPLAY command.

Adding Version Information

Introduction Using the Add a Tape Version panel, you can add a new version to an existing dataset definition or you can add new volume serial numbers to an existing dataset version.

Access To add a new tape version to a controlled dataset definition, access the Manage EDDs panel and enter A (for Add) to display the Add a Tape Dataset panel. To append reels to an existing version, access the Manage Versions panel and enter A (for Append) in the input field and press ENTER.

Add a Tape Version Panel

TSIO0203.t ---- BIM-EPIC - Add a Tape Dataset Version ----- USER=SJA
COMMAND ==>

Dataset ==> BILRECS BKUP TAPE

VERSION INFORMATION
Version Number ==> 1 (1 - 9999)
Version Block Size ==> (1 - 32767)
File Seq Number ==> (2 - 9999)
Offload Seq Number ==> (1 - 9999)
Use scratch tapes ==> (Y/N)

VOLSER(S)
1 2 3 4 5 More: +
7 8 9 10 11 12
13 14 15 16 17 18
19 20 21 22 23 24
25 26 27 28 29 30
31 32 33 34 35 36
37 38 39 40 41 42
43 44 45 46 47 48

F1=Help F3=Return F5=Recall F8=Fwd F11=Disk Add F12=Exit

Step	Action
1	Specify the desired attributes and volsers for the new version.
2	Press ENTER to add the tape version.

Using TSIDMNT ADD BIM-EPIC also provides the ADD command of the TSIDMNT catalog management utility to add a version to a controlled dataset. See Chapter 3, “DSN Catalog Management Utility”, for information on using the ADD command.

Locking a Version

Introduction

Locking a version prevents its use as input. You can lock a version from any panel that allows general version management.

Access

Enter the dataset name and Level V on the Search Catalog panel, and press ENTER to display the Manage Versions panel.

Manage Versions Panel

```
TSIO0200.D ----- BIM-EPIC - Manage Versions ----- USER=SJA
COMMAND ==>>
More: +>

Dataset Name          Version Number-of First  --- Creation ---
Width: 25             Vols/Exts Volser  Date           Time

_ BATCH CONVERT OUT          1          1    DUMMY6 08/01/1997 18:20
_ BILL BACKUP                1          1    DUMMY1 08/01/1997 19:51
_ BILL BACKUP                2          1    DUMMY1 07/31/1997 20:57
_ BILL BACKUP                3          1    DUMMY1 07/30/1997 19:23
_ BILL BACKUP                4          1    DUMMY3 07/29/1997 19:30
_ BILL.REPRINT.FILE          1          1    DUMMY3 08/01/1997 11:27
_ BILL.REPRINT.FILE          2          1    DUMMY6 07/31/1997 15:05
_ BILLING.PAN.ORDER          1          1    DUMMY0 08/01/1997 17:55
_ BILRECS BKUP TAPE          1          1    000050 07/29/1997 14:14
_ BILRECS BKUP TAPE          2          1    000024 06/26/1997 14:20
_ BILRECS BKUP TAPE          3          1    000061 05/28/1997 16:42
_ BILRECS BKUP TAPE          4          1    000019 04/29/1997 09:57
_ BILRECS BKUP TAPE          5          1    000143 03/28/1997 09:00

A = Append C = Close  E = Ext/Reels F = Release H = Retain I = Information
J = Lock   K = Unlock M = Mcatalog  S = Scratch T = Transfer Y = Cycle
F1=Help  F3=Return  F5=Recall  F8=Fwd  F11=Right  F12=Exit
```

Procedure

Step	Action
1	Place the cursor on the selection line of the version you wish to lock.
2	Type J (Lock) on the line and press ENTER.

Using TSIDMNT LOCK

BIM-EPIC also provides the LOCK command of the TSIDMNT catalog management utility to lock a version. See Chapter 3, “DSN Catalog Management Utility”, for information on using the LOCK command.

Releasing a Locked Version

Introduction Releasing a version permits one use of a locked version as input. You can release a locked version from any panel that allows general version management.

Access Enter the dataset name and Level V on the Search Catalog panel, and press ENTER to display the Manage Versions panel.

Manage Versions Panel

TSIO0200.D ----- BIM-EPIC - Manage Versions -----						USER=SJA
COMMAND ===>						
						More: +>
Dataset Name	Version	Number-of	First	---	Creation	---
Width: 25		Vols/Exts	Volser	Date		Time
- BATCH CONVERT OUT	1	1	DUMMY6	08/01/1997	18:20	
- BILL BACKUP	1	1	DUMMY1	08/01/1997	19:51	
- BILL BACKUP	2	1	DUMMY1	07/31/1997	20:57	
- BILL BACKUP	3	1	DUMMY1	07/30/1997	19:23	
- BILL BACKUP	4	1	DUMMY3	07/29/1997	19:30	
- BILL.REPRINT.FILE	1	1	DUMMY3	08/01/1997	11:27	
- BILL.REPRINT.FILE	2	1	DUMMY6	07/31/1997	15:05	
- BILLING.PAN.ORDER	1	1	DUMMY0	08/01/1997	17:55	
- BILRECS BKUP TAPE	1	1	000050	07/29/1997	14:14	
- BILRECS BKUP TAPE	2	1	000024	06/26/1997	14:20	
- BILRECS BKUP TAPE	3	1	000061	05/28/1997	16:42	
- BILRECS BKUP TAPE	4	1	000019	04/29/1997	09:57	
- BILRECS BKUP TAPE	5	1	000143	03/28/1997	09:00	
A = Append C = Close E = Ext/Reels F = Release H = Retain I = Information						
J = Lock K = Unlock M = Mcatalog S = Scratch T = Transfer Y = Cycle						
F1=Help F3=Return F5=Recall F8=Fwd F11=Right F12=Exit						

Step	Action
1	Place the cursor on the selection line of the version you wish to lock.
2	Type F (Release) on the line and press ENTER.

Using TSIDMNT RELEASE BIM-EPIC also provides the RELEASE command of the TSIDMNT catalog management utility to allow one use of a locked version for input. See Chapter 3, “DSN Catalog Management Utility”, for information on using the RELEASE command.

Unlocking a Version

Introduction

Unlocking a version allows its unrestricted use as input. You can unlock a version from any panel that allows general version management.

Access

Enter the dataset name and Level V on the Search Catalog panel, and press ENTER to display the Manage Versions panel.

Manage Versions Panel

TSIO0200.D ----- BIM-EPIC - Manage Versions ----- USER=SJA
COMMAND ==>>

More: +>

Dataset Name Width: 25	Version	Number-of Vols/Exts	First Volser	--- Creation --- Date	Time
_ BATCH CONVERT OUT	1	1	DUMMY6	08/01/1997	18:20
_ BILL BACKUP	1	1	DUMMY1	08/01/1997	19:51
_ BILL BACKUP	2	1	DUMMY1	07/31/1997	20:57
_ BILL BACKUP	3	1	DUMMY1	07/30/1997	19:23
_ BILL BACKUP	4	1	DUMMY3	07/29/1997	19:30
_ BILL.REPRINT.FILE	1	1	DUMMY3	08/01/1997	11:27
_ BILL.REPRINT.FILE	2	1	DUMMY6	07/31/1997	15:05
_ BILLING.PAN.ORDER	1	1	DUMMY0	08/01/1997	17:55
_ BILRECS BKUP TAPE	1	1	000050	07/29/1997	14:14
_ BILRECS BKUP TAPE	2	1	000024	06/26/1997	14:20
_ BILRECS BKUP TAPE	3	1	000061	05/28/1997	16:42
_ BILRECS BKUP TAPE	4	1	000019	04/29/1997	09:57
_ BILRECS BKUP TAPE	5	1	000143	03/28/1997	09:00

A = Append C = Close E = Ext/Reels F = Release H = Retain I = Information
J = Lock K = Unlock M = Mcatalog S = Scratch T = Transfer Y = Cycle
F1=Help F3=Return F5=Recall F8=Fwd F11=Right F12=Exit

Procedure

Step	Action
1	Place the cursor on the selection line of the version you wish to lock.
2	Type K (Unlock) on the line and press ENTER.

Using TSIDMNT UNLOCK

BIM-EPIC also provides the UNLOCK command of the TSIDMNT batch management utility to unlock a version for unrestricted use as input. See Chapter 3, “DSN Catalog Management Utility”, for information on using the UNLOCK command.

Closing a Version

Introduction It may be necessary to manually close a dataset version when it is in open status or conditionally cataloged by the abnormal termination of a job. You can close a version from the Manage Versions panel.

Access Once you have used the Search Catalog panel to enter your search criteria, the Manage Versions panel is displayed, listing all dataset versions that meet the name criteria specified.

Manage Versions Panel

TSIO0200.D ----- BIM-EPIC - Manage Versions -----						USER=SJA
COMMAND ===>						
						More: +>
Dataset Name	Version	Number-of	First	---	Creation	---
Width: 25		Vols/Exts	Volser	Date		Time
- BATCH CONVERT OUT	1	1	DUMMY6	08/01/1997	18:20	
- BILL BACKUP	1	1	DUMMY1	08/01/1997	19:51	
- BILL BACKUP	2	1	DUMMY1	07/31/1997	20:57	
- BILL BACKUP	3	1	DUMMY1	07/30/1997	19:23	
- BILL BACKUP	4	1	DUMMY3	07/29/1997	19:30	
- BILL.REPRINT.FILE	1	1	DUMMY3	08/01/1997	11:27	
- BILL.REPRINT.FILE	2	1	DUMMY6	07/31/1997	15:05	
- BILLING.PAN.ORDER	1	1	DUMMY0	08/01/1997	17:55	
- BILRECS BKUP TAPE	1	1	000050	07/29/1997	14:14	
- BILRECS BKUP TAPE	2	1	000024	06/26/1997	14:20	
- BILRECS BKUP TAPE	3	1	000061	05/28/1997	16:42	
- BILRECS BKUP TAPE	4	1	000019	04/29/1997	09:57	
- BILRECS BKUP TAPE	5	1	000143	03/28/1997	09:00	
A = Append C = Close E = Ext/Reels F = Release H = Retain I = Information						
J = Lock K = Unlock M = Mcatalog S = Scratch T = Transfer Y = Cycle						
F1=Help F3=Return F5=Recall F8=Fwd F11=Right F12=Exit						

Step	Action
1	Place the cursor on the selection line of the version you wish to close.
2	Type C (Close) on the line and press ENTER.

Using TSIDMNT CLOSE BIM-EPIC also provides the CLOSE command of the TSIDMNT catalog management utility to close a dataset version. See Chapter 3, “DSN Catalog Management Utility”, for information on using the CLOSE command.

Removing Version Information from the DSN Catalog

Introduction

A dataset's version information can be removed from the DSN Catalog by deleting its dataset definition or by purging its tape volume serial number.

Procedure

See “Deleting a Dataset Definition” or “Purging a Tape Volume” for information on actions which delete version information.

Using TSIDMNT DELETE and PURGE

BIM-EPIC also provides the DELETE and PURGE commands of the TSIDMNT catalog management utility to remove a dataset definition or purge a tape volume from the DSN Catalog. See Chapter 3, “DSN Catalog Management Utility”, for information on using the DELETE and PURGE commands.

Managing Tapes

Tapes in the DSN Catalog

Every tape used for a BIM-EPIC-controlled dataset is recorded in the DSN Catalog. Tapes are identified in the DSN Catalog by volume serial number. New tapes are added to the DSN Catalog automatically when BIM-EPIC-controlled datasets residing on those tapes are processed. You can also enter tape information into the DSN Catalog by using the Online Manager or the TSIDMNT ADD command. Volumes entered into an ACL are added to the dataset name '***.VOLUME.DSN.**' if they do not already exist.

Valid Volsers

Every controlled tape used must have a unique volume serial number. Even unlabeled tapes must be identified with their own volume serial numbers. This strict identification simplifies tape control and retrieval. For controlled datasets opened for output, BIM-EPIC checks the volume serial number of any new tape against the DSN Catalog and requires you to enter a new one or use a different tape if there is a duplication and still-active datasets on the tape.

Volume serial numbers can be any combination of alphanumeric characters, up to six characters in length. However, volume serial number 000000 is reserved by BIM-EPIC. It is used as a special marker in the DSN Catalog and cannot be used. Groups of volsers do not have to be consecutive. Any unique volume serial number is valid. Since tape pools are defined by ranges of volume serial numbers, it is helpful to initialize pool tapes with consecutive volume serial numbers.

How BIM-EPIC Produces Scratch Tapes

Scratch tapes are produced by the SCRATCH functions of the Online Manager and TSIDMNT. In addition, the TSIDUTL SCRATCH function produces a report listing all available scratch tapes. See 2-46 in this manual for more information. When a tape is scratched in the DSN Catalog, it is automatically scratched in the ACL also.

Tape Pooling

You can use tape pooling to group similar types of tapes into tape pools. For instance, your high quality tapes could constitute one pool, short tapes another, long tapes a third, and so on. Datasets are assigned to tape pools through the DSN Catalog. When a pooled dataset is created, BIM-EPIC requests the operator for a scratch tape from the appropriate pool.

Each pool is given a one-character pool code (A to Z or 0 to 9) and is assigned one or more volume serial number ranges. If a volume serial number falls within a pool's range, the tape belongs to that pool. A pool tape can only be used to write datasets assigned to that pool. If a volume serial number does not fall in any of the defined ranges, the tape is not a pool tape. It can only be used to write datasets which are not assigned to a tape pool. Datasets can be assigned to tape pools by manual or automatic cataloging.

If you use ACL support with pooling, the pools must be defined to the ACL system as "POOL-*x*" where *x* is the one-character BIM-EPIC pool code. Refer to the ACL vendor's documentation for information about tape pools.

How BIM-EPIC Selects Tapes for Mounting

Introduction

BIM-EPIC controls which tapes can be used for output. It checks the volume serial number of every tape processed. If the mounted volume is controlled, its serial number is compared with its corresponding DSN Catalog entry for scratch status before it can be used as output.

If the tape is uncontrolled, BIM-EPIC checks to see if the volume serial number is unique within the DSN Catalog. If the tape passes this test, it is added to the DSN Catalog and used automatically. If it does not, the tape is unloaded and a new tape requested.

Single or Prime Dataset Output

If the file has no restrictions, the operator can mount any valid scratch tape which does not belong to a tape pool.

If the file belongs to a tape pool, BIM-EPIC issues a general request for a scratch or a specific mount from that pool.

If revolving tapes are used, BIM-EPIC issues a general request for a scratched revolving tape from that dataset.

Subdataset Output

If the tape is not already mounted, BIM-EPIC requests the operator to mount the volume serial number of the last tape used for that multi-file group.

Uncontrolled Datasets

Ordinarily, no BIM-EPIC-controlled tape can be used to write an uncontrolled dataset.

ACL Support

If ACL support is activated, BIM-EPIC always attempts to satisfy a mount using an ACL volume. The ACL volume must meet the criteria of scratch status, mode, drive, availability, and so on, to be selected. If the ACL cannot satisfy the mount request, normal AVR is attempted, followed by the operator mount request message.

ACL cartridges should never be pre-mounted, unlike normal drives which can benefit from AVR.

Displaying Tape Volume Information

Introduction Online Manager provides the Tape Volume Information panel to display information on any controlled tape volume. You can also execute tape functions from the Tape Volume Information panel.

Access Choose M (for MANAGE) on the Primary Option Menu to display the Search Catalog panel. Type the desired volume serial number in the Volume Serial Number field. Press ENTER to display the Tape Volume Information panel.

Tape Volume Information Panel

```
TSIO0401.V ----- BIM-EPIC - Tape Volume Information ----- USER=SJA
COMMAND ==>

( _ ) Volser ==> 000143

VOLUME HISTORY
Expire Date: 03/28/1997 Create Date: 03/28/1997 ATTRIBUTES
Last Acc Date: 03/28/1997 Create Time: 09:00 Scratched: N
Last Acc Job: TVMBRRST Create Job: TVMBRBKP Unlabeled: N
Clean Date: . CPU ID: 9 Open: N
SYSLOG ID: C1 Revolving: N
CondCat: N

VOLUME INFORMATION
Reel: 1 Use Count: 6 LOCATION
Tape Pool: NA Error Count: NA Vault: 0
Device Addr: 0610 Media Type: CART Slot: 0
Tape Size: LARGE Tape Mode: F8 ACL Name: .
Blocks: 9987 ACL Cell: .

Current Dataset: .
First DS on Volume: BILRECS BKUP TAPE

C = Clean M = Move P = Purge
F1=Help F3=Return F5=Recall F12=Exit
```

Maintaining Tape Volume Information

Introduction

Use the Manage Tape Volumes panel to manage tape volume information.

Access

Choose M (for MANAGE) on the Primary Option Menu to display the Search Catalog panel. Enter the name of the dataset that owns the tape volume you want to manage, enter a 'V' in Level and press ENTER to display the Manage Versions panel. Type an 'E' to the left of the line containing the tape volume you wish to manage and press ENTER to display the Tape Volume Information panel.

Manage Tape Volumes Panel

```
TSIO0400.r ----- BIM-EPIC - Manage Tape Volumes ----- USER=SJA
COMMAND ==>>

Dataset: BILRECS BKUP TAPE                                More: NO
                                                           Version: 5

  Volser Reel  Creation   Expire   --- Last Access --- Use Vault Slot
           Date    Date    Date      Job      Count
-----
_ 000143     1 03/28/1997 03/28/1997 03/28/1997 TVMBRRST     6     0     0

C = Clean  I = Information  M = Move  P = Purge
F1=Help  F3=Return  F5=Recall  F12=Exit
```

Procedure

Both the Manage Tape Volumes and Tape Volume Information panels provides access to the following functions:

- CLEAN
- MOVE
- PURGE

Step	Action
1	You will be able to purge this volser from the Manage tape Volumes panel. NOTE: If you want BIM-EPIC to confirm all PURGE requests, make sure that the PURGE field is set to 'Y' in the Modify Profile Defaults (CONFIRM) panel.
2	Press ENTER to execute the requested function.

TSIDMNT

BIM-EPIC also provides the TSIDMNT catalog management utility to manage tape volumes. See 3-1 in this manual for information on TSIDMNT.

Purging a Tape Volume

Introduction You can purge a tape volume from the DSN Catalog by using the Manage Tape Volumes panel.

Access Choose M (for MANAGE) on the Primary Option Menu to display the Search Catalog panel. Enter the name of the dataset that owns the tape volume you want to manage, enter a 'V' in Level and press ENTER to display the Manage Versions panel. Type an 'E' to the left of the line containing the tape volume you wish to manage and press ENTER to display the Tape Volume Information panel.

Manage Tape Volumes Panel

```
TSIO0400.r ----- BIM-EPIC - Manage Tape Volumes ----- USER=SJA
COMMAND ==>

Dataset: BILRECS BKUP TAPE                                     More: NO
                                                                Version: 5

  Volser Reel  Creation   Expire   --- Last Access --- Use Vault Slot
           Date    Date    Date    Job    Count

_ 000143     1 03/28/1997 03/28/1997 03/28/1997 TVMBRRST      6    0    0

C = Clean  I = Information  M = Move  P = Purge
F1=Help  F3=Return  F5=Recall  F12=Exit
```

Step	Action
1	Place a P in the input area to the left of the Volser field and press ENTER.
2	If prompted, type YES to confirm the purge.

Using TSIDMNT PURGE BIM-EPIC also provides the PURGE command of the TSIDMNT catalog management utility to remove tape volume information from the DSN Catalog. See Chapter 3, “DSN Catalog Management Utility”, for information on using the PURGE command.

Recording Tape Cleaning

Introduction

You can record tape cleaning using either the Tape Volume Information or Manage Tape Volumes panels.

Access

Choose M (for MANAGE) on the Primary Option Menu to display the Search Catalog panel. Enter the name of the dataset that owns the tape volume you want to manage, enter a 'V' in Level and press ENTER to display the Manage Versions panel. Type an 'E' to the left of the line containing the tape volume you wish to manage and press ENTER to display the Tape Volume Information panel.

Tape Volume Information Panel

```
TSIO0400.r ----- BIM-EPIC - Manage Tape Volumes ----- USER=SJA
COMMAND ==>

Dataset: BILRECS BKUP TAPE                                     More: NO
                                                                Version: 5

  Volser Reel  Creation      Expire      --- Last Access --- Use Vault Slot
           Date       Date       Date       Job       Count

_ 000143     1 03/28/1997 03/28/1997 03/28/1997 TVMBRRST      6    0    0

C = Clean  I = Information  M = Move  P = Purge
F1=Help  F3=Return  F5=Recall  F12=Exit
```

Procedure

Step	Action
1	Place a C in the input area to the left of the Volser field and press ENTER.
2	Press ENTER to record tape cleaning.

You can also record tape cleaning from the Tape Volume Information panel. Place C in the input area in front of the Volser field and press ENTER.

Using TSIDMNT CLEAN

BIM-EPIC also provides the CLEAN command of the TSIDMNT catalog management utility to record tape cleaning. See Chapter 3, “DSN Catalog Management Utility”, for information on using the CLEAN command.

Retention and Scratching

Understanding Retention Concepts

Introduction

Retention is how long a dataset is kept before it can be discarded. Retention is one of the most critical areas of information control. Scratching, or making a tape available for reuse, is the physical storage consequence of the expiration of a tape dataset's retention. All automated tape management systems must ensure that retention is monitored and controlled in order to protect information assets.

How BIM-EPIC Implements Retention

BIM-EPIC maintains a variety of retention criteria. Retention can expire only when all criteria are met. This implementation protects your datasets against inadvertent scratching.

There are two BIM-EPIC functions for removing datasets: DELETE and SCRATCH. Single-file tape volumes associated with scratched dataset versions become valid scratch tapes immediately after using either of these functions. Multi-file tape volumes become valid scratch tapes when all associated datasets are scratched.

The DELETE Function

Use the DELETE function to remove a dataset definition and all its versions from the DSN Catalog and from BIM-EPIC control. All tape volumes associated with the dataset definition become scratch tapes according to the rules for single-file and multi-file tape volumes and are listed in the DSN Catalog under the dataset name '** DSN DELETED **'. See the section on "Managing Dataset Definitions" for further information.

The SCRATCH Function

Use the SCRATCH function to scratch versions of a specific dataset. Associated tape volumes can be used immediately as output tapes if all datasets on the affected tape have been scratched. BIM-EPIC resequences version numbers as you scratch each version. So if you scratch version number 1, what used to be 2 becomes 1, 3 becomes 2, and so on.

Initiating the Scratch Process

You must run a scratch before BIM-EPIC expires a dataset or allows a tape to be reused. Scratches can be performed by manual catalog management, using either the Online Manager or the TSIDMNT utility.

The TSIDUTL scratch process searches the entire DSN Catalog for all dataset versions which have satisfied their retention criteria and expires them. However, you can limit the scratch to certain user IDs or system IDs.

Setting Dataset Definition Retention

Introduction

When a dataset definition is created manually using the Define an EDD panel, a new set of retention criteria is defined which will apply to all future versions of that dataset.

Access

Select D (for DEFINE) from the Primary Option Menu to display the Define an EDD panel.

Define an EDD Panel

```
TSIO0103.D ----- BIM-EPIC - Define an EDD ----- USER=SJA
COMMAND ==>

DEFINE

Dataset ==> _____
Model   ==> _____

TAPE INFORMATION      DISK INFORMATION      DATASET INFORMATION
TPAdev: N  Mode:  ___  Disk Pool   ==>  _____  GEN or WRK   ==>  ___
Tape Pool  ==>  ___  Primary     ==>  _____  Retention   ==>  ___
Vault Meth ==>  ___  Secondary   ==>  _____  Cycles      ==>  ___
Owner ID   ==>  ___  Max Sec     ==>  _____  DLA         ==>  ___
Unlabeled  ==>  ___  Primary ALR ==>  _____  BLKSZ( _ CISZ) ==>  ___
Tape Size  ==>  ___  Auto Purge  ==>  _____  SYSnnn      ==>  ___
Revolving  ==>  ___                                     System ID   ==>  ___
                                           User ID     ==>  ___
                                           Password    ==>  ___
Prime DSN  ==>  _____  Encrypt Key  ==>  ___

F1=Help  F3=Return  F5=Recall  F12=Exit
```

Procedure

Enter the desired criteria to create the new dataset definition.

Using TSIDMNT CATALOG

BIM-EPIC also provides the CATALOG command of the TSIDMNT catalog management utility to set dataset definition retention. See Chapter 3, “DSN Catalog Management Utility”, for information on using the CATALOG command.

Modifying Dataset Definition Retention

Introduction When an existing dataset definition is modified using the Update an EDD panel, the existing retention criteria can be changed. The changes will apply to all subsequent versions of that dataset.

Access Select M (for MANAGE) from the Primary Option Menu to display the Search Catalog panel. After you have entered your search criteria, press ENTER to display the Manage EDDs panel. Place the cursor in front of the dataset definition you wish to modify and type U (for UPDATE). Press ENTER to display the Update an EDD panel.

Update an EDD Panel

TSIO0104.e ----- BIM-EPIC - Update an EDD ----- USER=SJA

COMMAND ==>

UPDATE

Dataset: BATCH CONVERT OUT

TAPE INFORMATION		DISK INFORMATION		DATASET INFORMATION	
TPAdev: N	Mode: NO	Disk Pool	==> DISK02	GEN or WRK	==> GEN
Tape Pool	==>	Primary	==> 50	Retention	==> 0
Vault Meth	==> 0	Secondary	==> 25	Cycles	==> 1
Owner ID	==>	Max Sec	==> 255	DLA	==> 0
Unlabeled	==> N	Primary ALR	==>	BLKSZ(_ CISZ)	==> OPT
Tape Size	==> L	Auto Purge	==> N	SYSnmm	==> 0
Revolving	==> N			System ID	==>
				User ID	==>
Comment	==>			Password	==>

F1=Help F3=Return F5=Recall F12=Exit

Procedure To modify the dataset definition, change the values in the desired input fields.

Using TSIDMNT UPDATE BIM-EPIC also provides the UPDATE command of the TSIDMNT catalog management utility to modify dataset definition retention. See Chapter 3, “DSN Catalog Management Utility”, for information on using the UPDATE command.

Setting or Modifying Dataset Version Retention

Introduction

Use the Retain a Version panel to set or modify the expiration date, creation date, and creation job name of a version. If the version is not in active status, it is put in active status automatically.

Access

From the Manage Versions panel, place an H (for Retain) next to a dataset name and press ENTER. The dataset name and version number to be updated are displayed on the Retain a Version panel.

Retain a Version Panel

```
TSIO0210.R ----- BIM-EPIC - Retain a Version ----- USER=SJA
COMMAND ==>

      RETAIN

      Tape Dataset: BILRECS BKUP TAPE
      Version:      1

      to

      Expire Date  ==> _____ (YYYYDDD format)
      Days + Today ==> _____ (1 to 999 days)
      Create Date  ==> _____ (YYYYDDD format)
      Create Job   ==> _____ (1 to 8 characters)
      TPA Device   ==> _____ (Y or N)
      Mode         ==> _____ (Creation Density)

      F1=Help  F3=Return  F5=Recall  F12=Exit
```

Procedure

Step	Action
1	Update any desired fields to modify dataset version retention:
2	Press ENTER. All data entered are verified for syntactic correctness and defaults added. Review the data before proceeding.
3	Press ENTER again to update the version retention or press PF3 to exit without updating.

Using TSIDMNT RETAIN

BIM-EPIC also provides the RETAIN command of the TSIDMNT catalog management utility to set and modify dataset version retention. See Chapter 3, “DSN Catalog Management Utility”, for information on using the RETAIN command.

Cycling a Dataset Version

Introduction

Cycling is a special purpose function for overriding the number of versions (cycles) retained for a dataset. Use it to take one version out of the normal retention cycle and reserve it for special retention. The version you cycle is:

- Removed from cycle retention for the dataset (it no longer counts as a version in the cycle)
- Retained until the expiration date you specify

For example, if you have told BIM-EPIC to keep three versions (cycles) of a particular dataset and you cycle one version, BIM-EPIC maintains four versions (the cycled version plus three normal versions) until the cycled version reaches its expiration date. All uncycled versions previous to the version you have just cycled are retained by cycle control only. Days Retention, Days Last Accessed Retention retention no longer apply. If the dataset is not retained under cycle control, those versions are eligible for scratch at the next run of TSIDUTL. Versions which have been cycled in the past are retained until their expiration date. Cycling a dataset has no effect on versions created after the cycled version. Cycling also allows you to change the creation date and the creation job name. These two fields have no effect on file retention.

From the Online Manager, the cycling function is executed using the Cycle a Version panel.

Difference Between CYCLE and RETAIN

Cycling looks very similar to the retain function. However, its effect is quite different. Using the retain function only affects the version you modify. Using the cycling function affects the version you modify and all previous versions.

Access

From the Manage Versions panel, place a Y next to a version and press ENTER.

Cycle a Version Panel

```

TSIO0211.Y ----- BIM-EPIC - Cycle a Version ----- USER=SJA
COMMAND ==>

        CYCLE

Tape Dataset: BILRECS BKUP TAPE
Version:      1

        to

Expire Date  ==> _____ (YYYYDDD format)
Days + Today ==> _____ (1 to 999 days)
Create Date  ==> _____ (YYYYDDD format)
Create Job   ==> _____ (1 to 8 characters)

F1=Help  F3=Return  F5=Recall  F12=Exit

```

Procedure

Step	Action
1	Update any desired fields to cycle the version.
2	Press ENTER. All data entered are verified for syntactic correctness and defaults added. Review the data before proceeding.
3	Press ENTER again to cycle the version or press PF3 to exit without cycling.

Using TSIDMNT CYCLE

BIM-EPIC also provides the CYCLE command of the TSIDMNT catalog management utility to cycle dataset versions. See Chapter 3, “DSN Catalog Management Utility”, for information on using the CYCLE command.

Scratching a Dataset Version

Introduction

There are several different status types that affect when a dataset version becomes eligible for scratch. Datasets in open status are eligible for scratch only after they have been closed with the Online Manager. All other versions are eligible for scratch only after all of the retention criteria specified for the dataset version has been satisfied. Scratching a dataset version using the Online Manager or the TSIDMNT SCRATCH command expires a specific dataset version you name. That version is scratched whether it has satisfied its retention criteria or not.

Access

Select M (for MANAGE) from the Primary Option Menu to display the Search Catalog panel. Enter the search criteria for the EDDs you wish to maintain, and press ENTER to display the Manage Versions panel. From this list, type V in front of the EDD whose versions you want to maintain. Press ENTER to display the Manage Versions panel.

Manage Versions Panel

TSIO0200.D ----- BIM-EPIC - Manage Versions -----						USER=SJA
COMMAND ==>>						
						More: +>
Dataset Name		Version	Number-of	First	--- Creation ---	
Width: 25			Vols/Exts	Volser	Date	Time
_ BATCH CONVERT OUT		1	1	DUMMY6	08/01/1997	18:20
_ BILL BACKUP		1	1	DUMMY1	08/01/1997	19:51
_ BILL BACKUP		2	1	DUMMY1	07/31/1997	20:57
_ BILL BACKUP		3	1	DUMMY1	07/30/1997	19:23
_ BILL BACKUP		4	1	DUMMY3	07/29/1997	19:30
_ BILL.REPRINT.FILE		1	1	DUMMY3	08/01/1997	11:27
_ BILL.REPRINT.FILE		2	1	DUMMY6	07/31/1997	15:05
_ BILLING.PAN.ORDER		1	1	DUMMY0	08/01/1997	17:55
_ BILRECS BKUP TAPE		1	1	000050	07/29/1997	14:14
_ BILRECS BKUP TAPE		2	1	000024	06/26/1997	14:20
_ BILRECS BKUP TAPE		3	1	000061	05/28/1997	16:42
_ BILRECS BKUP TAPE		4	1	000019	04/29/1997	09:57
_ BILRECS BKUP TAPE		5	1	000143	03/28/1997	09:00
A = Append C = Close E = Ext/Reels F = Release H = Retain I = Information						
J = Lock K = Unlock M = Mcatalog S = Scratch T = Transfer Y = Cycle						
F1=Help F3=Return F5=Recall F8=Fwd F11=Right F12=Exit						

Procedure

Place an S (for Scratch) in front of the version you want to scratch and press ENTER.

Using TSIDMNT SCRATCH

BIM-EPIC also provides the SCRATCH command of the TSIDMNT catalog management utility to scratch dataset versions. See Chapter 3, “DSN Catalog Management Utility”, for information on using the SCRATCH command.

3

DSN Catalog Management Utility

BIM-EPIC provides the TSIDMNT utility program for managing the DSN Catalog. The same functions can also be performed using the BIM-EPIC Online Manager.

Chapter Contents

Command Summary	3-3
BIM-EPIC System Requirements.....	3-4
BIM-EPIC Requirements.....	3-4
CA-EPIC for CMS Requirements.....	3-5
MNT Commands.....	3-6
Command Syntax.....	3-6
PASSWORD	3-7
ADD	3-8
CAPACITY	3-11
CATALOG	3-12
CLEAN	3-18
CLOSE.....	3-19
CYCLE	3-20
DELETE	3-22
DEQ.....	3-23
DISPLAY	3-24
ENQ.....	3-27
LOCK	3-28
MCATALOG.....	3-29
MOVE	3-30
OFFLINE (VSE Only).....	3-31
ONLINE (VSE Only)	3-32
PURGE	3-33
QUIESCE (VSE Only)	3-34
RELEASE.....	3-35
RENAME	3-36
RETAIN	3-37

SCRATCH.....	3-39
TRANSFER.....	3-41
UNLOCK.....	3-42
UPDATE	3-43

Command Summary

Command	Function
ADD	Add a version of a controlled dataset to the DSN Catalog
CAPACITY	Display DSN Catalog utilization statistics
CATALOG	Define a new EDD in the DSN Catalog (in Online Manager, use the DEFINE command)
CLEAN	Reset tape usage, error counts, and clean date
CLOSE	Change a version's status from "open" or "conditionally cataloged" to "active"
CYCLE	Override the cycle control of a dataset
DELETE	Remove a dataset and all of its versions from the DSN Catalog (tapes are set to scratch status)
DEQ	Dequeue the DSN Catalog
DISPLAY	Display DSN Catalog entries (EDDs, versions, and volumes)
ENQ	Enqueue the DSN Catalog
LOCK	Lock an EDD for output or a version from input
MCATALOG	Specify that a tape dataset can be accessed by more than one dataset name
MOVE	Change the vault and slot location of a tape volume
OFFLINE	Prevent a disk volume from being accessed (VSE only)
ONLINE	Enable access to a disk volume (VSE only)
PASSWORD	Specify or reset system password for catalog update
PURGE	Remove an EDD or versions from the DSN Catalog
QUIESCE	Make a disk volume ineligible for use in creating controlled datasets (VSE only)
RELEASE	Enable a locked EDD or version to be used once
RENAME	Change the name of a controlled dataset
RETAIN	Change a version's expiration date, creation date, or creation job, and make the version active
SCRATCH	Scratch a version of a dataset
TRANSFER	Move a version from one dataset to another
UNLOCK	Unlock a locked EDD or version
UPDATE	Change the attributes of a BIM-EPIC dataset definition (EDD)

BIM-EPIC System Requirements

BIM-EPIC Requirements

JCL Requirements

TSIDMNT is designed for use within production job streams, to scratch datasets as necessary and perform other periodic catalog management such as MCATALOG or CYCLE. TSIDMNT functions are submitted as shown below:

```
[// OPTION LOG]
// EXEC TSIDMNT
control-statements
.
/*
```

Print Options

Print output is controlled using the JCL LOG option. If the LOG option is active, all processed maintenance is printed on SYSLST. If the NOLOG option is in effect, only invalid commands are printed. If your system log option is NOLOG (the usual default), an // OPTION LOG statement is required prior to // EXEC TSIDMNT if a list of submitted commands is desired.

CA-EPIC for CMS Requirements

In order to use TSICMNT commands in CA-EPIC for CMS, link to the minidisk containing CA-EPIC for CMS user programs (for example, TSICSERV 191), and issue the following command:

```
GLOBAL TXTLIB TSICTXT
```

TSIDMNT Commands

Command Syntax

Description

TSIDMNT functions follow this syntax:

command [*'dataset-name'*] [*parameter-expressions*]

The command must always be first. It can be specified in full or abbreviated to its first three letters. For example, you can use either DISPLAY or DIS for the DISPLAY command.

dataset-name is positional. When used, it must always follow the command and be enclosed in apostrophes.

parameter-expressions can follow in any order, except where noted otherwise. A parameter-expression consists of a parameter and its value(s) that are joined by an equal sign (=). Parameter expressions must be separated from each other by a comma or blank spaces. Numeric values are automatically zero-filled on the left if required, so it is not necessary to use leading zeros.

**BIM-EPIC
Statements**

Statements can be entered in columns 1 through 71, and a statement can span up to three (3) cards. To continue a statement, place a non-blank character in column 72.

Example

ADD 'INVENTORY.FILE',
SER='IN0192,IN2323,IN2987,IN9870,IN4573,IN1029,IN3049' X

PASSWORD

Description

If your installation has used the PASSWD configuration option to invoke DSN Catalog update protection, you must use the PASSWORD command to enter the required password. The DSN Catalog password is required for all commands except the following:

- CAPACITY
- DISPLAY
- HELP
- PASSWORD

Submit the PASSWORD statement before any protected operation. The password is not printed on SYSLST if the command name (PASSWORD or PAS) begins in column 1. The password is cleared when the program ends.

Example

```
PAS LOLLIPOP
```

ADD

Description

Use the ADD command to add a new version to a controlled dataset. To create a new BIM-EPIC dataset definition (EDD) for a previously uncontrolled dataset, use the CATALOG command.

ADD can be used to add a new version that was created outside BIM-EPIC control or to recatalog a deleted version. Once a dataset is controlled, all new versions created under BIM-EPIC are added to the DSN Catalog automatically.

BIM-EPIC resequences version numbers as you add new versions. For example, if you add a new version, the previous version 1 becomes 2, 2 becomes 3, and so on. To add more than one version in a specific sequence, make sure to add the oldest version first and the newest version last.

It is possible for some versions of a tape dataset to have standard labels, while other versions are unlabeled. When you add a tape version to the DSN Catalog, its entry automatically receives the label type in the EDD. If a particular tape's label type differs from the EDD's, use the UPDATE command to modify the EDD's label type temporarily. Use ADD to add the new tape version. Then change the EDD back to the previous label type.

If an Automated Cartridge Library (ACL) is supported, ADD sets the tape volume to "non-scratch" status in the ACL inventory.

**BIM-EPIC
WARNING!**

If the VTOC entry for a disk dataset has been deleted, ADD does not re-create it. As long as the VTOC entry is deleted, that disk space is free and can be re-used for another dataset. If this occurs, do the following:

Ensure that the disk space has not already been reused.

Use the BIM-EPIC PACKSCAN ADD function to re-create the VTOC entry as soon as possible.

Use the ADD command to add the version to the DSN Catalog.

Syntax

```
ADD  'dataset-name' {SER=volser1,volser2,...,volser23
                    {ESER=volser1,volser2,...,volsern
                    {VOL=volser,start-track,tracks,...
                    {EVOL=volser,start-track,tracks,...
    [ ,VER=n ]
    [ ,GEN=n ]
    [ ,FIL=n ]
    [ ,BLK=n ]
    [ ,OFL=1 ]
```

ADD Parameters

Parameter	Required?	Valid Entry	Default
<i>dataset-name</i>	Yes	1 to 44 characters	None
Specify only one of the following two (required):			
SER (or ESER)		1 to 6 characters	None
VOL (or EVOL)		1 to 6 characters	None
Specify only one of the following two (optional):			
VER		1 to 999	1
GEN		1 to 999999	None
FIL	No	2 to 255	None
BLK	No	1 to 65535 MAX OPT	None
OFL	No	1	None

Descriptions

dataset-name specifies the name of the dataset to which the new version belongs.

SER/ESER (Tape versions only) specifies the tape volume serial number(s) on which this version is written. Specify the volume serial numbers of multi-volume datasets in reel sequence order. Use the ESER parameter instead of SER when a version is written on more than 23 tapes. You can also use ESER to add tapes to a version that is already in the DSN Catalog.

VOL/EVOL (VSE disk versions only) specifies the extents the dataset occupies. *volser* specifies the volume serial number of the volume that contains the extent.

start-track specifies the address at which the extent starts.

blocks specifies the number of blocks or tracks used for the extent.

VOL subparameters are positional. If there are multiple extents, repeat the subparameters in extent sequence. Use EVOL instead of VOL if the version has more than 10 extents. Also use it to add extents to a version that is already in the DSN Catalog. Supply extent information just as for VOL.

NOTE: For both VOL and EVOL, TSIDMNT does not access the disk volume(s) to determine whether the version exists or whether its extents match the extents you specify.

VER can be used to assign a specific version number for the dataset. If VER is not specified, VER=1 is assumed.

GEN can be used to specify the absolute generation number for the dataset. A generation number assigned to an existing version is not allowed, but you can specify a previously used generation number from a version that has been deleted. To assign a new (never used) generation number, it must be greater than the next generation number due to be created. For

example, if the highest current generation number (usually version 1) is 000120, specify a new generation number of 000122 or greater. The new version is always version 1.

FIL specifies the file sequence number if this version of a tape dataset is not the first dataset in a multi-dataset group.

BLK specifies the block size of this version of the dataset, if you wish. **MAX** sets the block size to track capacity for disk datasets and for tape datasets - 64K for VSE/ESA 2.5 and later, or 32K for earlier VSE releases. **OPT** sets the block size to half track for disk and for tape - 32K for VSE/ESA 2.5 and later, or 16K for earlier VSE releases.

OFL=1 can be used to identify the version as an offload version. While we do not recommend purging offload datasets, the **OFL** parameter can be used to assign offload status to a disk dataset that has been added to the catalog.

Examples

This example adds version number 3 to the PAYROLL.MASTER dataset. Version 3 is on tapes PY1234 and PY2764, with a block size of 10000.

```
ADD 'PAYROLL.MASTER',VER=3,BLK=10000,SER=PY1234,PY2764
```

This example adds version number 1 to the INVENTORY.237 dataset. Version 1 is on tape E93837.

```
ADD 'INVENTORY.237',SER=E93837
```


CAPACITY

Description Use the CAPACITY control statement to display DSN Catalog usage. This is displayed as both the number of active/free records, and as a percentage.

Syntax CAPACITY (no parameters)

Results The following is an example of the CAPACITY display.

EP119	TOTAL	CATALOG UTILIZATION			INDEX UTILIZATION		
EP119	CAPACITY	ACTIVE	FREE	PCT	ACTIVE	FREE	PCT
EP119	4560	959	3601	21	560	4000	12

CATALOG

Description

Use the CATALOG command to create a new BIM-EPIC dataset definition (EDD) in the DSN Catalog, and to define the attributes to be associated with that dataset name. After the EDD has been created, you can catalog versions of the dataset with the ADD command.

Syntax

CATALOG *'dataset-name'*
 [,AGP=YES|NO]
 [,ALR=n]
 [,BLK=n|MAX|OPT]
 [,CIZ=n]
 [,COM=*'comment'*]
 [,COP=*'existing-dataset-name'*]
 [,CYC=n]
 [,DEC=NO|*password*]
 [,DLA=n]
 [,EX2=n]
 [,EXT=n]
 [,LBL=YES|NO]
 [,MOD=<see mode table below>]
 [,NSU=*'prime-dataset-name'*]
 [,OWN=xx]
 [,POL=*poolname*]
 [,PWD=*password*]
 [,RET=n|PERM]
 [,REV=NO|YES]
 [,SEC=NO|n]
 [,SID=NO|xx]
 [,SIZ=SMALL|LARGE]
 [,SYSnnn]
 [,TPA=NO|YES]
 [,TPL=x|NO]
 [,UID=NO|xx]
 [,VLT=n]
 [,WRK=NO|YES]

General Parameters

Parameter	Required?	Valid Entry	Default
<i>dataset-name</i>	Yes	1 to 44 characters	None
BLK	No	NO 1 to 65535 MAX OPT	NO
COM	No	1 to 30 characters	None
COP	No	1 to 44 characters	None
CYC	No	0 to 999	None
DLA	No	1 to 9999	None
PWD	No	1 to 8 characters	NO
RET	No	1 to 9999 PERM	None
SID	No	NO 2 characters	NO
SYS <i>nnn</i>	No	000 to 253	000
UID	No	NO 2 characters	NO
WRK	No	YES NO	NO

Descriptions

dataset-name is used to locate a BIM-EPIC dataset definition (EDD) in the DSN Catalog. It must be unique. Generation dataset names can be a maximum of 35 characters long.

VSE Only: When adding CPU and partition-independent datasets, the dataset name must contain the specific CPU and partition IDs. For example, to use the dataset name '===.SORTWK1' on a DLBL in a job that will run on CPU 4 in the BG partition, specify the dataset name 'BG4.SORTWK1'.

BLK specifies the maximum output file blocksize to be used. When the file is opened for output, the actual blocksize used will be the largest possible value not exceeding the BLK value. This parameter is mutually exclusive with the Disk parameter CIZ. If both are specified for the same dataset definition, the last specified will prevail.

COM can be used to make notes about a dataset's use. The comment can be deleted by specifying COM=''

COP can be used to define a new dataset that is similar to a previously cataloged dataset. You must specify the COP parameter immediately following the dataset name. Any other parameters specified before the COP parameter are ignored. To make modifications to the copied attributes, you can specify additional parameters after the COP parameter.

PWD can be used to protect the dataset from unauthorized UNLOCK or RELEASE. If you specify NO, no password protection is provided. This parameter does not lock the dataset. To do so, use the LOCK command.

RET, CYC, DLA, WRK (Retention Days, Cycles Retention, DLA Retention, and Generation or Work) work together to determine dataset retention. These are some of the most important attributes of any dataset, and they must be carefully selected.

RET specifies the number of days to retain the dataset after its creation date.

CYC specifies the number of cycles (versions) to retain.

DLA specifies the number of days to retain the dataset after its last access date.

WRK specifies whether this is a generation dataset (WRK=NO) or a non-generation dataset (WRK=YES).

SID and **UID** (VSE Only) can be used to group your dataset with other datasets for catalog functions. If you specify NO, the ID is inactive.

SYSnnn (VSE Only) specifies a logical unit number to be used for the dataset, if you wish. Using this parameter inhibits LUB allocation automatically. If you specify SYS000, the parameter is deactivated.

Tape Parameters

Parameter	Required?	Valid Entry	Default
DEC	No	1 to 8 characters	None
LBL	No	YES NO	YES
MOD	No	<see mode table below>	None
NSU	No	1 to 17 characters	None
OWN	No	1 to 2 characters	None
REV	No	YES NO	NO
SIZ	No	LARGE SMALL	LARGE
TPA	No	YES NO	NO
TPL	No	1 character	NO
VLT	No	1 to 255	None

DEC is used to enter an encryption password if you want tape versions to be encrypted at creation. The encryption password cannot be added, changed, or deleted once it is entered in a BIM-EPIC dataset definition. This is not a password, and you will not be asked to re-enter it.

LBL specify NO for unlabeled, YES for standard labeled.

MOD specify the tape density. The following densities are supported for 3480/3490 devices:

- 00 Buffered write mode
- 08 Buffered write mode with IDRC
- 20 Unbuffered write mode
- 28 Unbuffered write mode with IDRC

The following densities are supported for 8809/9347 devices:

- 30 High speed / short gap
- 50 Low speed / long gap
- 60 Low speed / short gap
- 90 High speed / long gap

The following densities are supported for 3424/9348 devices:

- 42 Buffered write mode 1600 bpi
- 62 Unbuffered write mode 1600 bpi
- C2 Buffered write mode 6250 bpi
- E2 Unbuffered write mode 6250 bpi

The following densities are supported for 3410/3420/3422/3430 devices:

- D0 6250 bpi
- C0 1600 bpi
- C8 800 bpi

The following densities are supported for 3490E devices:

- F0 Buffered write mode (3490E)
- F8 Buffered write mode with IDRC (3490E)

Modes F0 and F8 are used internally by BIM-EPIC to distinguish 3490E tape cartridge drives from 3480/3490. Because VSE/ESA does not support these modes directly, BIM-EPIC sends mode 00 or 08 to the 3490E hardware whenever modes F0 or F8 are specified.

The following densities are supported for 3590 TPA devices:

- 00-07 Buffered model dependent formats
- 08-0F Buffered compressed model dependent formats
- 20-27 Unbuffered, like densities 00-07
- 28-2F Unbuffered, like densities 08-0F

Density 08 should be used for performance reasons.

NSU specifies that tape versions of the dataset are part of a multi-dataset group. Specify the name of first dataset on the tape (the prime dataset).

Enter the prime dataset name when the dataset is first cataloged. BIM-EPIC does not allow you to add a prime dataset name to an existing BIM-EPIC dataset definition (EDD)..

OWN specifies an Owner ID, if you wish. If you also specify a tape pool, make sure the Owner ID falls within the tape pool range. If you specify NO, no Owner ID is used.

REV regulates tape selection and can be used for backup tapes. Revolving tapes and datasets have an exclusive relationship. A revolving tape volume can only be used for its dataset, and a revolving tape dataset can only be written on tape volumes assigned to it. A revolving tape dataset cannot be used in an Automated Cartridge Library (ACL).

SIZ specifies a tape size.

TPA can be specified as YES or NO. Specify NO for non-TPA devices, YES for TPA devices. IF MOD is set for a TPA device, TPA=YES must be coded before the MOD parameter.

TPL specifies a tape pool ID.

VLT specifies a vault method number, if you want tape vaulting for this dataset.

Disk Parameters

Disk information is only meaningful to BIM-EPIC, but it can be updated from CA-EPIC for CMS.

Parameter	Required?	Valid Entry	Default
AGP	No	Yes or NO	NO
ALR	No	1 to 99999	None
CIZ	No	NO 512 to 32768	None
EXT	No	1 to 999999	DEFEXT
EX2	No	0 to 999999	50% of primary
POL	No	1 to 6 characters	None
SEC	No	0 to 255	255

AGP specifies whether dataset retention is controlled by the Automatic Generation Purge feature.

ALR specifies the number of records to be written to a version, if you wish. Specifying allocation by logical records overrides the primary extent size specified by the EXT parameter if both parameters are specified.

CIZ specifies the desired control interval size to be used when allocating on an FBA DASD. This parameter should be avoided for files allocating on CKD DASD as its use will defeat reblocking. The value specified will be adjusted to an even multiple of 512. This parameter is mutually exclusive with the parameter BLK. If both are specified for the same dataset definition, the last specified will prevail.

EXT specifies the primary extent size in blocks or tracks. The minimum size is 3 tracks for CKD devices, or the value specified in the CIZ parameter for FBA devices. If the DEFEXT configuration option is set to **YES** for your installation, you can use that as a default primary extent, if you wish.

EX2 specifies the secondary extent size, if you wish. Remember that the secondary extent defaults to 50% of the size of the primary extent. The minimum extent size is 3 tracks for CKD devices, or the value specified in the CIZ parameter for FBA devices.

POL specifies a disk pool name if disk versions are to be created. Use disk pool names defined in the TSIDPOL macro. If the DEFEXT configuration option is set to **YES** for your installation, use that as a default disk pool, if you wish.

SEC specifies the maximum number of secondary extents to be allowed, if you wish. NO (the default) allows 255 secondary extents. To inhibit all secondary allocations, specify zero (0).

Examples

The following example catalogs the PAYROLL.MASTER dataset, copying dataset attributes from the 'AP.MASTER' dataset and changing the dataset password to PASSWORD.

```
CATALOG 'PAYROLL.MASTER', COP='AP.MASTER', PWD='PASSWORD'
```

The following example catalogs the BALANCE.BUDGET dataset, with day retention of 3 and days last accessed retention of 3. Disk datasets are written in disk pool POOLA1 with a primary extent size of 20 and a maximum of 3 secondary extents.

```
CAT 'BALANCE.BUDGET',RET=3,DLA=3,POL=POOLA1,EXT=20,SEC=3
```

The following example results in BIM-EPIC retaining three versions of the BALANCE.BUDGET dataset at all times. When a new version is created, the oldest version is automatically deleted at end-of-job.

```
CAT 'BALANCE.BUDGET',CYC=3,AGP=YES...
```

CLEAN

Description

Use the CLEAN command to record tape cleaning. CLEAN sets the use and error counts for cleaned tapes to zero.

Syntax

`CLEAN SER=volser1,volser2,...`

Parameters

Parameter	Required?	Valid Entry	Default
<i>volsern</i>	Yes	1 to 6 characters	None

SER specifies the volume serial numbers of the cleaned tapes.

Example

The following example records the cleaning of tapes A38472, A94874, and 857594

`CLEAN SER=A38472,A94874,857594`

CLOSE

Description

BIM-EPIC only accesses active versions of a dataset. Datasets that are in “open” or “conditionally cataloged” status cannot be processed, and they become eligible for scratch differently from active versions. To access or save an open or conditionally cataloged dataset, you must close it with the CLOSE command.

CLOSE only changes the DSN Catalog status of a version. It does not perform close processing, such as writing tape or disk labels.

If an Automated Cartridge Library (ACL) is supported, CLOSE sets the tape volume to “non-scratch” status in the ACL inventory.

Syntax

```
CLOSE  'dataset-name' [ ,SER=volser ] [ ,GEN=nnn ]
```

Parameters

Parameter	Required?	Valid	Default
<i>dataset-name</i>	Yes	1 to 44 characters	None
SER	No	1 to 6 characters	None
GEN	No	1 to 999999	None

dataset-name specifies the name of the dataset.

SER specifies the volume serial number of a tape version.

GEN specifies the generation number of a disk dataset.

If neither SER or GEN is specified, BIM-EPIC closes the most recently created version of *dataset-name*.

Example

The following example closes the BATCH.BALANCE dataset.

```
CLOSE  'BATCH.BALANCE'
```

CYCLE

Description

The CYCLE command is a special purpose command for overriding the normal Cycles Retention (CYC) of a dataset. The CYCLE command causes one version to be removed from the Cycles Retention and reserved for special retention. For example, you might use it to isolate the correct balance sheet from a series of trial balance runs.

CYCLE looks very similar to the RETAIN command. However, its effect is quite different. Using RETAIN only affects the version you modify. Using CYCLE affects the version you modify and all previous versions.

The version you CYCLE is:

- removed from cycle retention for the dataset (that is, it no longer counts as a version in the cycle).
- retained until the expiration date you specify.

For example, if the dataset has CYC=3 retention and you CYCLE one version, BIM-EPIC maintains four versions (the cycled version plus three normal versions) until the cycled version reaches its expiration date.

All uncycled versions created before the version that is cycled are retained by cycle control only. Retention days and DLA retention no longer affect cycled versions. If the dataset does not have cycle control, uncycled versions are eligible for scratch.

Versions which have been cycled in the past are retained until their expiration date.

The CYCLE command also allows you to change the creation date and the creation job name. These two fields have no effect on dataset retention.

Syntax

```
CYCLE  'dataset-name' , { VER=n
                        , SER=volser
                        , GEN=n
                        }
                        { , EXD=yyyyddd
                        , DAY=n
                        , CDT=yyyyddd
                        , JOB=new-name }
```

Parameters

Parameter	Required?	Valid Entry	Default
<i>dataset-name</i>	Yes	1 to 44 characters	None
Specify only one of the following three (required):			
VER		1 to 999	None
SER		1 to 6 characters	None
GEN		1 to 999999	None
Specify at least one of the following parameters (required):			
EXD		1900001 to 2099366	None
DAY		1 to 999	None
CDT		1900001 to 2099366	None
JOB		1 to 8 characters	None

dataset-name specifies the dataset name.

Specify either VER, SER, or GEN:

VER specifies the version number.

SER specifies the volume serial number of the first (or only) volume of a tape version.

GEN specifies the generation number of a disk version.

At least one of the following parameters must be included in the command statement. Specify dates (EXD and CDT) in Julian format (4-digit year followed by 3-digit day, for example January 1, 1991 = 1991001).

EXD specifies the new expiration date of the version that was cycled. If neither an expiration date nor period retention is specified, the version is retained according to the period retention established in the dataset defaults.

DAY specifies period retention. BIM-EPIC calculates the expiration date as today's date plus the number of days entered.

CDT specifies a new creation date. CDT has no effect on dataset expiration.

JOB changes the name of the job that created this version. JOB has no effect on dataset expiration.

Example

The following example cycles the AGED.TRIAL.BALANCE dataset, setting apart version 3 with a 7-day retention and changing its creating job name to TRIALBAL. All versions earlier than version 3 (for example, versions 4–9) are now retained by cycle retention only.

```
CYCLE 'AGED.TRIAL.BALANCE', VER=3, DAY=7, JOB=TRIALBAL
```

DELETE

Description Use the DELETE command to remove a dataset and all its versions from the DSN Catalog and from BIM-EPIC control. The BIM-EPIC dataset definition (EDD) is deleted. All tapes associated with the dataset become scratch tapes and are listed under the dataset name ‘**DSN DELETED **’ on BIM-EPIC reports and in Online Manager displays. Disk space allocated to the dataset is freed.

If an Automated Cartridge Library (ACL) is supported, DELETE sets the tape volume to “scratch” status in the ACL inventory.

WARNING! Encrypted datasets must not be deleted from the DSN Catalog if you ever plan to use them again as input. Once an encrypted version's EDD is deleted, it cannot be decoded and is forever unreadable.

Syntax `DELETE 'dataset-name'`

Parameter	Required?	Valid Entry	Default
<i>dataset-name</i>	Yes	1 to 44 characters	None

dataset-name specifies the dataset name to be deleted.

Example This example deletes the dataset AP.BATCH.BALANCE.TEST.

```
DEL 'AP.BATCH.BALANCE.TEST'
```

DEQ

Description

Use DEQ to dequeue the BIM-EPIC DSN Catalog after it has been enqueued with the ENQ command.

In CA-EPIC for CMS, this command is only meaningful when sharing a BIM-EPIC DSN Catalog.

Syntax

DEQ (no parameters)

DISPLAY

Description

Use the DISPLAY command to display entries in the DSN Catalog. The display lists all cataloged information about the dataset and each of its versions. The summary display contains one line of data for each version of each dataset.

If a version resides on multiple volumes, each volume has its own line in the display. Scratched tape versions continue to be displayed until their tapes are used to write other datasets.

For a detailed explanation of the displays, see the Appendix. Most of the items listed in the reports are self-explanatory. However the ST (file status) and the TYPE (file type) headings require immediate explanation.

A code for dataset status is displayed under the column heading ST. The valid status codes are:

- (blank) - active version
- C - conditional catalog status
- L - active version in a locked status
- M - MCAT version
- O - open status
- S - scratch status

There are three dataset types listed under the TYPE heading: TAPE, DISK, and EMPTY. TAPE and DISK indicate the media on which the dataset is written. EMPTY signifies that the dataset has been cataloged, but it contains no versions. A dataset is empty when it is first cataloged, before any versions are created. A dataset can also be empty if all its versions have been scratched and all tapes have been reused.

Syntax

```

DISPLAY { 'dataset-name'
          SER=volser
          ALL
          'string.ALL' }

[ ,ACTIVE ]
[ ,DSK | TAP ]
[ ,DSN ]
[ ,JOB=creation-jobname ]
[ ,NEW ]
[ ,POL=poolname ]
[ ,PULL ]
[ ,SCRATCH ]
[ ,SID=xx ]
[ ,UID=xx ]

```

Parameters

Parameter	Required?	Valid Entry	Default
Specify one of the following four (required):			
<i>dataset-name</i>		1 to 44 characters	None
SER		1 to 6 characters	None
ALL		ALL	None
<i>string.ALL</i>		1 to 44 characters	None
ACTIVE	No	ACTIVE	OFF
Specify only one of the following two (optional):			
DSK		DSK	OFF
TAP		TAP	OFF
ACTIVE	No	ACTIVE	OFF
DSN	No	DSN	OFF
JOB	No	1 to 8 characters	None
NEW	No	NEW	OFF
POL	No	1 to 8 characters	None
PULL	No	PULL	OFF
SCRATCH	No	SCRATCH	OFF
SID	No	2 characters	None
UID	No	2 characters	None

Primary Parameters

dataset-name requests a detailed display of the dataset named.

SER requests a detailed display of the contents of the specified tape volume.

ALL requests a summary display of all datasets and entries in the DSN Catalog. ALL is the default when you specify any optional parameters.

'*string.ALL*' requests a summary display of all datasets beginning with the specified character string.

**Secondary
Parameters**

The following parameters are valid with the ALL and string.ALL parameters.

ACTIVE displays only active versions.

DSK displays only disk versions.

DSN displays dataset names only (omits dataset summaries).

JOB displays the latest version of all datasets that were created by the specified jobname. It also displays any OPEN or conditionally cataloged datasets created by that job since the latest version was created.

NEW displays all datasets that have been created since the last execution of the TSIDUTL DSPLY NEW function.

POL displays only datasets from the specified disk pool.

PULL displays only the latest version. Place PULL after the ALL or string.ALL parameter, as in this example: DIS 'PB.PAY.ALL',PULL

SCRATCH displays only scratch tapes.

SID displays only those datasets with the specified system ID. If specified, the SID and UID parameters must be the last parameters in the command.

TAP displays only tape versions.

UID displays only those datasets with the specified user ID. If specified, the SID and UID parameters must be the last parameters in the command.

Example

The following example produces a summary display of all active datasets with the creation job name of AR8000.

```
DISPLAY ALL,ACTIVE,JOB=AR8000
```

The following example produces a summary display of all the scratched versions of datasets that begin with the character string SDR3.

```
DIS 'SDR3.ALL',SCRATCH
```

The following example displays a detailed display of the AP.BALANCE dataset.

```
DIS 'AP.BALANCE'
```


ENQ

Description

Use ENQ to enqueue the BIM-EPIC DSN Catalog and prevent access to it by jobs running in other partitions or CPUs. To enqueue a DSN Catalog that is shared by two or more CPUs, MULTCPU=YES must be specified as a configuration option.

In CA-EPIC for CMS, this command is only meaningful when sharing a BIM-EPIC DSN Catalog.

Syntax

ENQ (no parameters)

LOCK

Description

Use the LOCK command to lock a dataset or version.

If you do not specify a version, volume serial number, or generation number, the EDD is locked for output, which means that BIM-EPIC does not permit additional versions to be created until you release or unlock it. If you do specify a version, volume serial number, or generation number, only that version of the dataset is locked for input, which means that it cannot be read until you release or unlock it.

Syntax

```
LOCK  'dataset-name'  [ ,VER=n  
                      ,SER=volser  
                      ,GEN=n ]
```

Parameters

Parameter	Required?	Valid Entry	Default
<i>dataset-name</i>	Yes	1 to 44 characters	None
Specify only one of the following three (optional):			
VER		1 to 999	None
SER		1 to 6 characters	None
GEN		1 to 999999	None

dataset-name specifies the name of the dataset to be locked.

To lock a specific version, specify either VER, SER, or GEN:

- VER specifies the version number.
- SER specifies the volume serial number of the first (or only) volume of a tape version.
- GEN specifies the generation number of a disk version.

Example

The following example locks all versions of the PAYROLL.MASTER dataset.

```
LOCK  'PAYROLL.MASTER'
```

The following example locks only version 1 of the 'PAYROLL.MASTER' dataset.

```
LOC  'PAYROLL.MASTER',VER=1
```

MCATALOG

Description

Use the MCATALOG command to specify that a tape dataset can be accessed by more than one dataset name. The new dataset names are known as MCAT aliases. After a version has been MCATALOGed, it is accessible by both its original dataset name and its MCAT alias(es). The dataset is retained until all retention criteria from all associated BIM-EPIC dataset definitions (EDDs) have been satisfied. MCATALOG can be used to create special retention for a tape version. Subdatasets cannot be MCATALOGed. When a display of an MCATALOGed version is requested by volume serial number, the original dataset record is displayed.

Syntax

```
MCAT    'from-dataset-name' { ,VER=from-ver|SER=from-ser } ,
        'to-dataset-name' [ ,VER=to-ver ]
```

Parameters

Parameter	Required?	Valid Entry	Default
<i>from-dataset-name</i>	Yes	1 to 44 characters	None
One of the following two must be specified:			
<i>from-ver</i>		1 to 999	None
<i>from-ser</i>		1 to 6 characters	None
<i>to-dataset-name</i>	Yes	1 to 44 character	None
<i>to-ver</i>	No	1 to 999	1

MCAT parameters are positional.

from-dataset-name specifies the name of the original dataset.

You must specify either SER or VER:

from-ver specify the version number of the original dataset.

from-ser specify the volume serial number of the first (or only) volume of the original dataset.

to-dataset-name specifies the MCAT alias.

to-ver specifies the version number of the MCAT alias.

Example

Make version 2 of the PAYROLL.MASTER dataset to be also version 1 of the PAYROLL.PERM dataset, so it can be accessed using either dataset name.

```
MCA 'PAYROLL.MASTER' ,VER=2 , 'PAYROLL.PERM' ,VER=1
```

MOVE

Description

Use the MOVE command to change the vault and slot location of a tape volume.

If an Automated Cartridge Library (ACL) is supported and the tape is moved *from* the main vault (Vault 0), the volume is ejected from the ACL at the time the MOVE statement is processed.

WARNING!

Vaulting reports do not reflect changes specified by MOVE commands. If any dataset on the volume is controlled by automatic vaulting, the vaulting program will undo your MOVE and return the tape to its normal place in the rotation.

Syntax

MOVE SER=*volser*,*vault*,*slot*

Parameters

Parameter	Required?	Valid Entry	Default
<i>volser</i>	Yes	1 to 6 characters	None
<i>vault</i>	Yes	1 to 255	None
<i>slot</i>	Yes	1 to 9999	None

volser specifies the volume serial number of the tape to be moved.

vault specifies the number of the tape's new vault.

slot specifies the number of the tape's new slot.

vault and *slot* are positional parameters which must be entered in the sequence shown.

Example

The following example moves tape AP3847 to vault 6, slot 234.

```
MOV SER=AP3847,6,234
```

OFFLINE (VSE Only)

Description

Use the OFFLINE command to prevent a disk volume from being accessed by BIM-EPIC if you must dismount the volume or “device down” the drive. Once a volume is in OFFLINE status, BIM-EPIC will not use it to allocate either output or input datasets. OFFLINE cannot disable volume access across machines, so you must issue the OFFLINE command from all machines that have access to the device.

Use the ONLINE command to restore BIM-EPIC access to a volume that has been declared OFFLINE. See ONLINE and QUIESCE.

Syntax

OFFLINE *volser*

Parameters

Parameter	Required?	Valid Entry	Default
<i>volser</i>	Yes	1 to 6 characters	None

volser specify the serial number of the volume you wish to remove from BIM-EPIC access.

Example

The following example removes the disk volume SYSWK1 from BIM-EPIC access.

```
OFF SYSWK1
```

ONLINE (VSE Only)

Description

Use the ONLINE command to enable BIM-EPIC to access a disk volume for both input and output datasets. If a disk volume has been placed in OFFLINE or QUIESCE status, use ONLINE to restore access to the volume. ONLINE cannot enable a volume across machines, so you must use the ONLINE command in all machines in which the OFFLINE command was used for the volume.

See OFFLINE and QUIESCE.

Syntax

ONLINE *volser*

Parameters

Parameter	Required?	Valid Entry	Default
<i>volser</i>	Yes	1 to 6 characters	None

volser specify the serial number of the volume you want to place online.

Example

The following example gives BIM-EPIC access to disk volume SYSWK2.

```
ONL SYSWK2
```

PURGE

Description

Use the PURGE command to remove BIM-EPIC dataset definitions (EDDs), versions, and tape volume serial numbers from the DSN Catalog. With PURGE, you can delete an entire dataset and all its versions (tape and disk) or simply remove a tape volume and the versions it contains.

Datasets which are part of a multi-dataset group cannot be purged until all other datasets in the group are scratched or deleted. In addition, all versions must be in closed status before they can be purged.

If an Automated Cartridge Library (ACL) is supported, PURGE sets the tape volume to “scratch” status in the ACL inventory.

WARNING!

Do not use this command for routine deletion of datasets.

When a tape volume is purged, BIM-EPIC does not attempt to determine whether it is a valid scratch tape. For this reason, use this command only when a physical tape volume is destroyed or permanently removed from your installation.

Encrypted dataset versions must not be purged from the DSN Catalog. Once information about an encrypted version is deleted, it cannot be decoded and is forever unreadable.

Syntax

```
PURGE { 'dataset-name'
        { SER=volser1,volser2,... } }
```

Parameters

Parameter	Required?	Valid Entry	Default
Specify one of the following two (required):			
<i>dataset-name</i>		1 to 44 characters	None
SER		1 to 6 characters	None

dataset-name specifies a dataset name when you wish to delete an EDD and all associated versions from the DSN Catalog. The dataset is no longer controlled.

SER specifies a tape volume serial number when you wish to remove that serial number from the DSN Catalog, but leave the dataset under BIM-EPIC control. If the volume serial number contains non-alphanumeric characters, enclose it in apostrophes.

Examples

The following example purges PAYROLL.TABLE and all associated tape volumes.

```
PURGE 'PAYROLL.TABLE'
```

The following example purges volume PR3746 and the version written on it.

```
PUR SER=PR3746
```

QUIESCE (VSE Only)

Description

QUIESCE makes disk volumes ineligible for the creation of controlled datasets. BIM-EPIC will not allocate output datasets on a volume that has been quiesced. Quiesced volumes are still eligible for BIM-EPIC input processing and for uncontrolled processing. QUIESCE cannot disable a volume across machines, so you must use QUIESCE on all machines that have access to the volume.

Use the ONLINE command to remove a volume from quiesced status. Use the OFFLINE command to remove a volume from both input and output processing.

Syntax

`QUIESCE volser`

Parameters

Parameter	Required?	Valid Entry	Default
<code>volser</code>	Yes	1 to 6 characters	None

`volser` specify the serial number of the volume you want to quiesce.

Example

The following example quiesces the disk volume SYSWK1.

```
QUI SYSWK1
```


RELEASE

Description

Use the RELEASE command to unlock, for one use only, EDDs or versions that have been placed in LOCK status (the UNLOCK command allows unlimited access). If you do not indicate a specific version, the EDD is released for one use as output, so that you can create one more version of the dataset. The new version is placed in locked status automatically.

If a specific version is indicated, that version is released for input, and BIM-EPIC allows you to read it only once. After a released version is read, it is automatically put back into locked status. RELEASE requires prior use of the LOCK command.

Syntax

```
RELEASE 'dataset-name' [,specifier-expression] [,PWD=password]
```

Parameters

dataset-name is required and can be up to 44 characters long.

specifier-expression is optional. If used, it must be one of the following:

- VER=*version-number*
- SER=*volser*
- GEN=*generation-number*

where *version-number* is an integer from 1 to 999. *volser* identifies the first (or only) volume of a tape version and can be up to 6 characters. *generation-number* is an integer from 1 to 999999.

password must be used if the dataset is password-protected. It can be up to 8 characters.

Examples

The following example releases the PAYROLL.MASTER dataset, supplying its password HARRY, and permits one more version to be created.

```
RELEASE 'PAYROLL.MASTER', PWD=HARRY
```

The following example releases the version of the PAYROLL.MASTER dataset which starts on volume serial number PR9485, and permits that version to be read once as input.

```
REL 'PAYROLL.MASTER', SER=PR9485
```

RENAME

Description The RENAME command allows you to change the name of a cataloged dataset. All existing versions of the original dataset are referenced by the new dataset name, and all dataset characteristics remain the same.

 If the dataset is a disk dataset, its VTOC entries are also renamed.

WARNING! Do not rename prime datasets or offload datasets. Results are unpredictable.

Syntax `RENAME 'old-dataset-name' , 'new-dataset-name'`

Parameter	Required?	Valid Entry	Default
<i>old-dataset-name</i>	Yes	1 to 44 characters	None
<i>new-dataset-name</i>	Yes	1 to 44 characters	None

old-dataset-name specifies the name of the dataset you want to rename.

new-dataset-name specifies the new dataset name.

Example The following example changes the name of the dataset CURRENT.MASTER to APRIL.MASTER.

`REN 'CURRENT.MASTER' , 'APRIL.MASTER'`

RETAIN

Description

Use the RETAIN command to alter a version's expiration date, creation date, or creation job name. If the version is not in active status, it is put in active status automatically.

If an Automated Cartridge Library (ACL) is supported, RETAIN sets the tape volume to “non-scratch” status in the ACL inventory. RETAIN also changes the status of a version from scratch, open or conditional catalog to active status.

Syntax

```
RETAIN 'dataset-name', specifier-expression, modifier-expression[, OF2]
```

Parameters

dataset-name is required.

specifier-expression is required. It must be one of the following:

- VER=nnn where nnn is an integer from 1 to 999.
- SER=xxxxxx where xxxxxx identifies the first (or only) volume of a tape version and can be up to 6 characters.
- GEN=nnnnnn where nnnnnn is an integer from 1 to 999999.

modifier-expression is required. It must be one of the following:

- EXD=*expiration-date* where expiration date is the new expiration date of the version in the format yyyyddd.
- DAY=*retention-days* where retention days specifies period retention. BIM-EPIC calculates the expiration date as today's date plus the number of retention days.
- CDT=*creation-date* where creation date is the Julian date on which the version was created in the format yyyyddd.
- JOB=*jobname* where *jobname* is the name of the job that created this version of the dataset.
- TPA=xxx where xxx is YES for TPA devices, NO for non-TPA devices. If MOD is specified with a TPA density, then TPA=YES must be specified before the MOD parameter.
- MOD=xx where xx is the density that the version was created with.

OF2 is an optional keyword parameter. If used, it goes at the end of the RETAIN statement. It can be used to identify the RETAINED version as having FAVER2 offload format. While we do not recommend purging any offload datasets, the OF2 parameter can be used to assign FAVER2 offload format to a tape after it has already been ADDED to a dataset with the OFL parameter.

Examples

The following example retains version 3 of the AGED.TRIAL.BALANCE dataset. Version 3 is closed and is retained for seven days from today, and its creating job name is changed to TRIALBAL.

```
RETAIN 'AGED.TRIAL.BALANCE',VER=3,DAY=7,JOB=TRIALBAL
```

The following example retains the version of PAYROLL.UNEDITED.TRANSACTIONS which starts on volume PR5867. The version is closed and expires on November 23, 1997.

```
RET 'PAYROLL.UNEDITED.TRANSACTIONS',SER=PR5867,EXD=1997327
```

SCRATCH

Description

Use the **SCRATCH** command to scratch a version of a dataset. The version must be in active status to be scratched. If it is in open or conditionally cataloged status, you must first use the **CLOSE** command to make it active.

If an Automated Cartridge Library (ACL) is supported, **SCRATCH** sets the tape volume to “scratch” status in the ACL inventory.

In BIM-EPIC, you can use the **TSIDMNT SCRATCH** command in a job stream to scratch disk datasets without supplying the dataset’s password. As long as the current job name and partition are the same as the creating job’s name and partition, BIM-EPIC disregards its password protection.

WARNING!

Encrypted dataset versions must not be scratched. If an encrypted version is scratched, it cannot be decoded and is forever unreadable.

Syntax

```
SCRATCH 'dataset-name' { ,VER=n
                        ,SER=volser } [FOR]
                        ,GEN=n
```

Parameters

Parameter	Required?	Valid Entry	Default
<i>dataset-name</i>	Yes	1 to 44 characters	None
Specify only one of the following three (required):			
VER		1 to 999	None
SER		1 to 6 characters	None
GEN		1 to 999999	None
Specify the following optional keyword parameter only if the disk volume on which a dataset resides is no longer available:			
FOR	No	None	N/A

dataset-name specifies the dataset name.

You must specify either VER, SER, or GEN:

VER specifies the version number.

SER specifies the volume serial number of the first (or only) volume of a tape version.

GEN specifies the generation number of a disk version.

FOR specifies that the disk volume on which a dataset resides is no longer available and the entry is to be removed from the catalog .

Scratching by volume serial number is not valid for NSU subdatasets or MCAT aliases.

Examples

The following example scratches version 1 of the INVENTORY.SANDIEGO.3847 dataset.

```
SCRATCH 'INVENTORY.SANDIEGO.3847',VER=1
```

The following example scratches the version of the INVENTORY.SANDIEGO.3847 dataset which starts on volume IN4658.

```
SCR 'INVENTORY.SANDIEGO.3847',SER=IN4658
```

TRANSFER

Description

Use the TRANSFER command to move a version from one dataset to another. The version transferred is no longer associated with its original dataset name. A transferred version retains all of its original characteristics, except for version number.

VSE Only: If the version is a disk dataset, the associated VTOC entry is also renamed.

Syntax

```
TRANSFER 'from-dataset-name', from-specifier,  
         'to-dataset-name'[,VER=to-version]
```

Parameters

from-dataset-name is required.

from-specifier is required. It must be one of the following:

- VER=*version-number* where *version-number* is an integer from 1 to 999.
- SER=*volser* where *volser* is the serial number of the first (or only) volume of a tape version.

to-dataset-name is required.

to-version is optional. If specified, it must be an integer from 1 to 999.

Example

The following example transfers version 7 of the IRVINE.INVENTORY dataset to version 1 of the SANDIEGO.INVENTORY dataset.

```
TRA  'IRVINE.INVENTORY',VER=7,'SANDIEGO.INVENTORY'
```

UNLOCK

Description

Use the UNLOCK command to permanently unlock a locked dataset or version. To unlock a dataset or version for one use only, use the RELEASE command instead. Use of UNLOCK requires prior use of the LOCK command.

To permit unlimited access to a specific version, use UNLOCK with a version qualifier. To permit creation of an unlimited number of new versions, use UNLOCK without a version qualifier.

Syntax

```
UNLOCK  'dataset-name',dataset-specifier [,PWD=password]
```

Parameters

dataset-name is required.

dataset specifier is required. It must be one of the following:

- VER=*version-number*
- SER=*volser*
- GEN=*generation-number*

password must be used if the dataset has been password-protected.

Examples

The following example unlocks the PAYROLL.MASTER dataset, supplying its dataset password HARRY. Unlimited versions of the dataset can now be created.

```
UNLOCK  'PAYROLL.MASTER',PWD=HARRY
```

The following example unlocks version 6 of the INVENT.958575 dataset. It can now be read as input an unlimited number of times.

```
UNL  'INVENT.958575',VER=6
```


UPDATE

Description

Use UPDATE to change the characteristics associated with an EDD in the DSN Catalog. The updated characteristics will only affect versions created after the update. To update previously created versions, use CLOSE, CYCLE, or RETAIN.

The parameters available on the UPDATE command are nearly identical to the CATALOG command parameters. However, the prime dataset name and encryption password parameters cannot be updated once the dataset is cataloged. To change processing for a dataset once these parameters have been entered, you must create a new dataset name for it in your DSN Catalog and change existing JCL.

To remove the following parameters if they are already cataloged, set them to NO:

- disk pool
- password
- system ID
- tape pool
- user ID

Syntax

Refer to the CATALOG command earlier in this chapter for complete syntax rules.

Examples

The following example updates the PAYROLL.EDITED.TRANSACTIONS dataset, changing its vault method to number 3.

```
UPDATE 'PAYROLL.EDITED.TRANSACTIONS',VLT=3
```

The following example updates the SR3.INVENTORY.45768 dataset, changing its default block size to 10,000 and changing its default density to 1600 bpi.

```
UPD 'SR3.INVENTORY.45768',BLK=10000,MOD=3
```


4

Accessing the DSN Catalog

This chapter discusses several ways to access the DSN Catalog.

Chapter Contents

Displaying DSN Catalog Entries (TSIDUTL DSPLY)	4-2
Accessing DSN Catalog Records (TSIDAC2)	4-5
ADDRESS EPIC	4-11

Displaying DSN Catalog Entries (TSIDUTL DSPLY)

Description

The DSPLY command creates reports from the data maintained in the DSN Catalog. Reports can be created displaying data associated with all the datasets, all the version entries, specific datasets, or specific versions which match specified criteria. These reports can also be produced in different sort sequences.

DSPLY produces reports listing all datasets or groups of datasets in the DSN Catalog. In most cases, you receive a full report of each dataset's catalog entry. The BIM-EPIC dataset definition can be made bold by overprinting for easier viewing.

For report examples, see "TSIDUTL Report Examples" in the *BIM-EPIC Installation and System Guide*.

UPSI Values

Setting	Action
lxxx	Lists dates in European format (DD/MM/YY)
xlxxx	Suppresses dataset name overprinting. Use this setting if the report will only be viewed at a terminal or if the report will be printed on a printer that does not support overprinting.
xxxlx	Suppresses catalog update when using the NEW subparameter
xxxxl	Bypasses CA-EPIC for CMS Archive datasets

Dataset Requirements

SORTWK1 is required if you use the SEQ parameter.

Syntax

```
DSPLY dataset-qualifier [,report-qualifier]
      [,SEQ=sort-sequence]
```

Parameters

dataset-qualifier is required. It must be one of the following:

ACTIVE reports only active datasets.

ALL reports all datasets in the DSN Catalog.

DISK reports only on disk datasets.

DSN='dataset-name' reports only the specified dataset.

NEW reports dataset versions created since the last time the DSPLY NEW report was run. NEW can be used to track the datasets created within a given time frame. NEW updates the catalog so that at the next run it can distinguish the new datasets. If you do not want the update, execute TSIDUTL with UPSI xxx1x.

PULL reports the latest version of all datasets.

PULL,DSN='dataset-name' reports the latest version of the named dataset.

SCRATCH reports scratch tapes.

TAPE reports only on tape datasets.

TAPEPOOL reports the tapes in each tape pool. Page breaks occur between tape pools.

VAULT reports the tapes stored in each offsite vault. Page breaks occur between vaults.

report-qualifier is optional. If used, it must be one of the following:

Values	Reports Only...
OWNER	Datasets with the specified owner ID
SYSID	Datasets with the specified system ID
TPL	Tapes within the specified pool
USERID	Datasets with the specified user ID
VLT	datasets stored in the specified vault.

SEQ=sort-sequence defines the sort sequence of your report. Valid sort sequences are listed below. **SEQ=sort-sequence** is an optional parameter. If used, it must be the last parameter specified. If you do not use it, datasets are listed in the order they are found in the index. You might want to sort the index by using the INDEX function before running reports without the SEQ parameter.

If SEQ is used, sort-sequence must be one of the following:

Value	Sorts By...
CJOB	Creation jobname
CDATE	Creation date
DSN	Dataset name
ERROR	Total errors
SERIAL	Tape volume serial number
VAULT	Vault and slot numbers
TAPEPOOL	Tape pool ID
XDATE	Expiration date

Examples

The following job produces a report listing all datasets belonging to the System ID TS and the User ID MR. They are listed in the order they are found in the catalog's index.

```
// JOB DISPLAY DSN CATALOG
// EXEC TSIDUTL
DSPLY ALL,SYSID=TS,USERID=MR
/*
/ &
```

The following job produce a report listing all vaulted tapes. Tapes are listed in dataset name sequence within the vault.

```
// JOB DISPLAY DSN CATALOG
// ASSGN SYS001,150
// DLBL SORTWK1,'===.SORTWK1,(X)'
// EXTENT SYS001
// EXEC TSIDUTL
DSPLY VAULT,SEQ=DSN
/*
/ &
```

Accessing DSN Catalog Records (TSIDAC2)

Description

You may want to access DSN Catalog information with your own programs to produce your own reports or for other reasons. The TSIDAC2 subroutine is provided for this purpose and is located in the BIM-EPIC installation library as member name TSIDAC2.OBJ.. A sample COBOL program (TSIDACC2) is located in the BIM-EPIC installation library as member name TSIDACC2.C.

TSIDAC2 is not reentrant and should not be used in an online CICS environment.

DSN Catalog Record Structure

Each dataset's DSN Catalog entry consists of two types of records:

- A single dataset definition (EDD) record (called the **master record**).
- Individual version-level records (called **detail records**). One record exists for each version, whether it is valid, open, conditionally cataloged, or scratched. If the version is multi-volume, there is a detail record for each additional volume.

The DSN Catalog dataset also includes an index. Each time a new record is added to the DSN Catalog file, another index entry is added to the end of the index area. This means that information obtained sequentially may not be in alphabetical order. To re-sequence the index, use the INDEX function of TSIDUTL.

About Sequential and Random Access

In random access, TSIDAC2 returns the master record for a specific dataset the first time it is called for that dataset. On each subsequent call, TSIDAC2 returns a detail record for that dataset.

In sequential access, TSIDAC2 returns each record for a single dataset in order, and then moves to the next dataset in the DSN Catalog's index sequence and begins returning its records, one at a time. Detail records are always returned current version first, then each successively older version.

TSIDAC2 Calling Procedure

Description

TSIDAC2 returns DSN Catalog information for any program language or report writer.

TSIDAC2 is a relocatable module in the BIM-EPIC installation library. The library that contains TSIDAC2 must be available when you catalog your calling program.

The procedure for using the TSIDAC2 subroutine is:

Step 1

Define the storage area to be used for the parameter list during a TSIDAC2 call.

This storage area begins with a one-byte field that is used as a return code from the TSIDAC2 subroutine. The rest of the storage area contains the data being passed to and received from the TSIDAC2 subroutine.

The TSIDAC2.C and TSIDAC2.A copybooks contain the layout of the storage area, and are in the BIM-EPIC installation library. Use one of these copybooks to define the storage area.

Step 2

Place the appropriate argument in the USR-KEY field to tell TSIDAC2 what function to perform. Function calls are described in detail in the next section.

Step 3

Call TSIDAC2 from your program.

This call requires that a parameter list be passed to TSIDAC2. This parameter list consists solely of the storage area discussed in Step 1.

Step 4

Following the TSIDAC2 call, you should check the return code that is passed back in the USER-RETURN-CODE field.

Return Codes

The following return codes are used:

0 indicates that the requested function has been successfully completed and the data has been returned in USER-CATALOG-DATA.

1 indicates that end-of-file has been reached. This return code is issued when reading the entire DSN Catalog via Sequential Access or when reading all versions of a specific dataset using Random Access.

2 indicates that the requested data was not found in the DSN Catalog during a Random Access call.

3 indicates that the BIM-EPIC system is not active.

Data is returned only if the return code is 0.

Example

This is an example of the COBOL coding that can be used to invoke the TSIDAC2 subroutine.

```
WORKING-STORAGE SECTION.  
01 ACC-RECORD COPY TSIDAC2.  
PROCEDURE DIVISION.  
    MOVE (search argument) TO USR-KEY.  
    CALL 'TSIDAC2' USING ACC-RECORD.  
    IF USER-RETURN-OK GO TO PROCESS-THE-RECORD.  
    IF USER-RETURN-EOF GO TO END-OF-FILE.  
    IF USER-RETURN-NOT-ACTIVE GO TO SYSTEM-DOWN.  
    IF USER-RETURN-NOT-FOUND GO TO NOT-FOUND.  
PROCESS-THE-RECORD.
```

TSIDAC2 Search Arguments and Calls

Description

The following pages detail the specific procedures necessary to gain different types of access to the DSN Catalog using TSIDAC2. Implement these procedures as steps 2 and 3 of the procedure outlined in the previous section.

For the sake of discussion, we use the COBOL copybook (TSIDAC2.C) to name the data fields described in this section, and all examples are coded in COBOL.

Random Access

To request a complete dataset entry:

1. Move the dataset name to the USR-KEY field.
2. Issue multiple calls to TSIDAC2.

Example

```
PROCEDURE DIVISION.
.
.
.
DSN-LOOP.
    MOVE 'DATA.SET.NAME.A' TO USR-KEY.
    CALL 'TSIDAC2' USING ACC-RECORD.
    IF USER-RETURN-CODE = 1 GO TO END-OF-DSN
    (process the catalog information here)
    GO TO DSN-LOOP.
```

To request a master record only:

1. Move the dataset name to the USR-KEY field.
2. Call TSIDAC2 just once.

Example

```
PROCEDURE DIVISION.
.
.
.
    MOVE 'DATA.SET.NAME.A' TO USR-KEY.
    CALL 'TSIDAC2' USING ACC-RECORD.
.
.
.
```

To request a detail record by volume serial number:

1. Move a hex 02 to USR-KEY-POS-1 and move the six-digit volume serial number to USR-KEY-SER.
2. Call TSIDAC2.

Example

```
WORKING-STORAGE SECTION.
01 HEX-NUMBER    PIC S9(4) COMP VALUE +2.
01 HEX-REDEF REDEFINES HEX-NUMBER.
    05 FILLER    PIC X.
    05 HEX02     PIC X.
.
.
.
PROCEDURE DIVISION.
.
```

```

      .
      .
      MOVE HEX02 TO USR-KEY-POS-1.
      MOVE 'EP3452' TO USR-KEY-SER.
      CALL 'TSIDAC2' USING ACC-RECORD.
      .
      .
      .

```

Sequential Access

To read the entire DSN Catalog sequentially:

1. Move spaces to the USR-KEY field.
2. Issue multiple calls to TSIDAC2.

Example

```

PROCEDURE DIVISION.
      .
      .
      .
      SEQUENTIAL-READ.
      MOVE SPACES TO USR-KEY.
      CALL 'TSIDAC2' USING ACC-RECORD.
      IF USER-RETURN_CODE = 1
          GO TO END-OF-CATALOG.
      (process the catalog information here)
      GO TO SEQUENTIAL-READ.

```

To specify a starting point for sequential processing:

1. Move the starting dataset name into the USR-KEY field.
2. Call TSIDAC2.
3. Move spaces to the USR-KEY field.
4. Issue multiple calls to TSIDAC2.

Example

```

PROCEDURE DIVISION.
      .
      .
      .
      MOVE 'DATA.SET.NAME.B' TO USR-KEY.
      CALL 'TSIDAC2' USING ACC-RECORD.
      SEQUENTIAL-READ.
      MOVE SPACES TO USR-KEY.
      CALL 'TSIDAC2' USING ACC-RECORD.
      IF USER-RETURN_CODE = 1
          GO TO END-OF-CATALOG.
      (process the catalog information here)
      GO TO SEQUENTIAL-READ.

```

Terminating Sequential Access

Sequential access to the DSN Catalog is normally terminated at end-of-file. In certain circumstances, however, you may wish to terminate sequential processing prior to end-of-file and allow your programs to change between sequential and random processing. No records are returned when performing this type of call.

To terminate sequential processing prior to end-of-file:

1. Move high-values (hex FF) to the USR-KEY field.
2. Call TSIDAC2.

Example

```
PROCEDURE DIVISION.  
.  
.  
.  
MOVE HIGH-VALUES TO USR-KEY.  
CALL 'TSIDAC2' USING ACC-RECORD.  
.  
.  
.
```

Releasing GETVIS

The first time TSIDAC2 is called, it acquires partition GETVIS storage for various purposes. If you want to release this storage:

1. Move low-values (hex 00) to USR-KEY.
2. Call TSIDAC2.

Example

```
PROCEDURE DIVISION.  
.  
.  
.  
MOVE LOW-VALUES TO USR-KEY.  
CALL 'TSIDAC2' USING DSN-RECORD.  
.  
.  
.
```

ADDRESS EPIC

Description

The ADDRESS EPIC command provides an interface to BIM-EPIC from BIM's Global Subsystem (BIM-GSS) product.

Return Codes

Code	Explanation
0	Function completed normally
-1	ADDRESS EPIC command is invalid
-2	BIM-EPIC is not active
-4	Invalid keyword on ADDRESS EPIC command line
-5	Invalid keyword value for numeric operand
-6	STEM keyword is missing
-8	ID keyword is missing
-9	Period is missing from STEM operand
-10	STEM name is invalid
-11	Insufficient getvis for VBUF work area
-12	GRXEXCOM error return
-13	GETDSA failure

Command Forms

The following form returns the current BIM-EPIC release. The stem *n* will be in the format vv.rr.mm.

```
ADDRESS EPIC 'EXTRACT RELEASE STEM n.'
```

The following form returns the master record for the dataset specified by the variable *ii*. If the dataset name is coded as a literal and has embedded blanks, it must be enclosed in single quotes (') or double quotes ("). If stem *n.0* is null, the requested dataset was not found.

```
ADDRESS EPIC 'EXTRACT ID' ii 'STEM n.'
```

The following form returns the master record for datasets whose dataset name starts with the character string *ii*. The stem *n.0* indicates the total number of records returned as stem variables. If *n.0* is 0, there were no datasets found whose name starts with the specified character string *ii*. The COUNT keyword is required describing the maximum number of records to return. A specification of COUNT * will return all matching records. The SKIP keyword may optionally be used specify the number of records that will be bypassed before returning them as stem variables.

In the following examples, REXX coding rules apply, that is, if an argument is a literal, it is within quote marks; if it is a variable it is outside the quote marks. Two examples:

```
ADDRESS EPIC 'EXTRACT DSN ID' ii 'STEM n. COUNT' c 'SKIP' s
```

```
ADDRESS EPIC 'EXTRACT DSN ID' ii 'STEM n. COUNT 5 SKIP 8'
```

The following form returns the master record for all datasets in the BIM-EPIC catalog. The stem *n.0* indicates the total number of records returned as stem variables. If *n.0* is null, there were no datasets found. The COUNT keyword may optionally be used to specify the maximum number of records to return. A specification of COUNT *, the default when ID is *, will return all records. The SKIP keyword may optionally be used to specify the number of records that will be bypassed before returning them as stem variables.

```
ADDRESS EPIC 'EXTRACT DSN ID * STEM n. COUNT' c 'SKIP' s
```

The following form returns the detail record for the dataset *ii*. The stem *n.0* indicates the total number of records returned as stem variables. If *n.0* is null, there were no detail records found for the dataset *ii* or the datasets is not found in the BIM-EPIC catalog. The COUNT keyword may optionally be used to specify the maximum number of records to return. A specification of COUNT *, the default, will return all detail records associated with the datasets. The SKIP keyword may optionally be used to specify the number of records that will be bypassed before returning them as stem variables.

```
ADDRESS EPIC 'EXTRACT DETAIL ID' ii 'STEM n. COUNT' c 'SKIP' s
```

The following form returns the detail record for the tape volume *v*. If stem *n.0* is null, the requested volume serial was not found.

```
ADDRESS EPIC 'EXTRACT VOL' v 'STEM n.'
```

Processing Output

If you need to use data records extracted by the EXTRACT ID, EXTRACT DETAIL, or EXTRACT VOL instructions, they can be decoded by including the GREXX member EPICREC.OAL in your IMODs. This member is loaded into the MON PDS during Online Manager installation.

5

BIM-EPIC TLBL and DLBL Statements

This chapter describes BIM-EPIC's TLBL and DLBL statements. BIM-EPIC TLBL statements are slightly different from VSE TLBLs and provide additional options. BIM-EPIC DLBL statements have the same format as VSE DLBLs and also provide additional options. BIM-EPIC provides an MVS-style DD statement for VSE. See 6-1 in this manual for further information. Each type of BIM-EPIC label statement provides its own benefits. Which you choose depends largely on what you want to do with the dataset you are processing.

Chapter Contents

TLBL Statements	5-2
TLBL Options.....	5-6
DLBL Statements.....	5-8
DLBL Options	5-10

TLBL Statements

Introduction

BIM-EPIC provides modifications to the standard VSE TLBL statement. Similarities and differences are shown in the table below.

All TLBL fields are positional.

TLBL Format

Pos	BIM-EPIC Format and Terminology	VSE Format and Terminology
1	// TLBL <i>dtfname</i>	// TLBL <i>filename</i>
2	[, <i>dataset-name</i> [, <i>alpha-opts</i>]]	[, <i>fileid</i>]
3	[, <i>retention</i>]	[, <i>retention</i>]
4	[, <i>volser</i>]	[, <i>file-serial-number</i>]
5	[, <i>reel-number</i>]	[, <i>volume-sequence-number</i>]
6	[, <i>file-sequence-number</i>]	[, <i>file-sequence-number</i>]
7	[, <i>version-number</i>]	[, <i>generation-number</i>]
8	[, <i>numeric-options</i>]	[, <i>version-number</i>]
9	[, <i>disposition</i>]	[, <i>disposition</i>]

Parameters

Parameter	Required?	Valid Entry	Default
<i>dtfname</i>	Yes	1 to 7 characters	None
<i>dataset-name</i>	For output	See description below	Blanks
<i>alpha-opts</i>	No	See description below	None
<i>retention</i>	No	1 to 9999 or yyyy/ddd	None
<i>volser</i>	No	1 to 6 characters	None
<i>reel-number</i>	No	1 to 256	1
<i>file-sequence-number</i>	No	1 to 9999	None
<i>version-number</i>	No	1 to 999	1
<i>numeric-options</i>	No	0 to 99	0

dtfname

dtfname specifies the symbolic name that the program uses for the dataset, for example TAPEIN and SORTIN. IBM calls this field the *filename*.

Dataset Name Field

The **dataset name field** consists of *dataset-name* and any desired *alpha-opts*. A beginning and ending apostrophe are required. This parameter is required for output. For input, the dataset name field can be omitted and a volume serial number supplied instead. However, if you want BIM-EPIC to select the tape, specify this parameter. The maximum length of the dataset name field (including *dataset-name*, commas, options, and parentheses) is 58 characters.

dataset-name

dataset-name specifies the name of the actual dataset to be used. IBM calls this the *fileid*. For BIM-EPIC to control the dataset, *dataset-name* must be identical to an EDD name in the DSN Catalog. The maximum length of *dataset-name* for a prime dataset in a multi-file group is 17 characters. Single dataset or subdataset names can be up to 44 characters in length.

alpha-opts

alpha-opts are provided for compatibility with tape management products from other vendors. The JCLOPT configuration option must be YES to use these options. You can specify alphabetic codes, version requests, and density requests (in any order), as long as they are separated from *dataset-name* and each other by commas. See “TLBL Options” below for alphabetic codes. A specific version can be requested by specifying a parenthetical expression that contains a minus sign (-) followed by an integer that is one less than the version desired. For example, to retrieve catalog version 2, specify (-1).

The following densities are supported for 3480/3490 devices:

- 00 Buffered write mode
- 08 Buffered write mode with IDRC
- 20 Unbuffered write mode
- 28 Unbuffered write mode with IDRC

The following densities are supported for 8809/9347 devices:

- 30 High speed / short gap
- 50 Low speed / long gap
- 60 Low speed / short gap
- 90 High speed / long gap

The following densities are supported for 3424/9348 devices:

- 42 Buffered write mode 1600 bpi
- 62 Unbuffered write mode 1600 bpi
- C2 Buffered write mode 6250 bpi
- E2 Unbuffered write mode 6250 bpi

The following densities are supported for 3410/3420/3422/3430 devices:

- D0 6250 bpi

C0 1600 bpi
C8 800 bpi

The following densities are supported for 3490E devices:

F0 Buffered write mode (3490E)
F8 Buffered write mode with IDRC (3490E)

Modes F0 and F8 are used internally by BIM-EPIC to distinguish 3490E tape cartridge drives from 3480/3490. Because VSE/ESA does not support these modes directly, BIM-EPIC sends mode 00 or 08 to the 3490E hardware whenever modes F0 or F8 are specified.

retention	<i>retention</i> specifies the number of days to retain this version.
volser	<i>volser</i> requests a specific tape volume serial number. IBM calls this the file serial number. This field is optional, and <i>we recommend that it be omitted</i> .
reel-number	<i>reel-number</i> specifies the reel number to be processed first. Use this parameter to skip initial reels in a multi-volume group. If you specify a volume serial number, <i>reel-number</i> is ignored.
file-sequence-number	<i>file-sequence-number</i> specifies the file number of the dataset on the tape.
version-number	<i>version-number</i> requests a specific version. Use the number assigned to the version in the DSN Catalog for the version. If you are requesting version number 2, specify '2' here. If you specify a volume serial number, <i>version-number</i> is ignored. This parameter replaces the generation number in the VSE format.
numeric-options	<i>numeric-options</i> specifies numeric option codes. See “TLBL Options” below for the numeric option codes and their meanings. This parameter replaces the version number in the VSE format.
disposition	<i>disposition</i> specifies file disposition as DISP=OLD, DISP=NEW, or DISP=MOD. It is identical to the VSE parameter and is valid only for VSE releases 3.5 and above. Refer to IBM TLBL documentation for more information.
VSE LOGSRC Option	The VSE LOGSRC option has no effect on TLBL statements. BIM-EPIC performs the symbolic parameter substitution and will only display the TLBL after substitution, not before.

Examples

In the following example, BIM-EPIC selects the tape volume BACKUP.STR (version 1) from the DSN Catalog.

```
// TLBL TAPEIN, 'BACKUP.STR'
```

In the following example, BIM-EPIC selects the BACKUP.STR (version 3) from the DSN Catalog.

```
// TLBL TAPEIN, 'BACKUP.STR, (-2)'
```

In the following example, BIM-EPIC honors the request for volume EP6173.

```
// TLBL TAPEIN, , , EP6173
```

In the following example, BIM-EPIC selects BACKUP.STR, and specifies "Rewind and unload at close" and mode '08'.

```
// TLBL TAPEIN, 'BACKUP.STR,U,08'
```

TLBL Options

Description

TLBL option codes provide special BIM-EPIC processing that is unavailable with standard VSE TLBL parameters. See “TLBL Statements” above for instructions and restrictions on specifying parameters.

Alphabetic option codes (called **alpha options**) are provided for compatibility with other tape management products. Alpha options can only be specified within the TLBL dataset name field.

Numeric codes can only be specified in the TLBL *numeric-options* parameter. To specify multiple numeric codes, total the different codes together. For example, if you want to specify option 3 (Do not rewind) and option 64 (Inhibit LUB allocation), specify 67 in the numeric-options parameter.

Numeric options can be used for uncontrolled datasets.

Option Codes

Alpha	Numeric	Action
	0	No options in effect
W	1	Rewind at close
U	2	Rewind and unload at close
	3	Do not rewind
R	4	Scratch at close
H	8	Hold the drive assignment at close
D	16	Data set drop
	32	Conditional catalog
	48	Concatenating datasets
	64	Inhibit LUB allocation

0 (No Options in Effect)

When no options are coded, BIM-EPIC honors the rewind code in your program.

Rewind options in the TLBL override rewind specifications in your program.

1 or W (Rewind)

Rewinds the tape to load point at close. This eliminates operator intervention when the same tape is reused in another step or job. If a cartridge is being used on an ACL drive, BIM-EPIC unloads it at close unless the “Hold assignment at close” option (8 or H) is specified.

2 or U (Unload)	Rewinds and unloads the tape at close.
3 (Do not rewind)	Prevents rewinding the tape at close. Use this option when you are creating multi-dataset tapes.
4 or R (Scratch at close)	Scratches the dataset when it is closed as input. This option does not affect subdatasets.
8 or H (Hold the drive assignment at close)	Prevents BIM-EPIC from releasing the tape drive at close. If Autoattach is on, it also prevents the drive from being detached. The drive will be released at end-of-job, when the logical unit is assigned to another device, or when the tape drive is deviced down, whichever occurs first.
16 or D (Dataset Drop)	Use this option when your program issues multiple "open" requests for the same DTF name. While VSE alone would process the original label each time the same DTF name is opened with the drop option, BIM-EPIC processes the next label in the JCL with that DTF name. (Dataset drop differs from concatenation, since concatenation processes multiple datasets with one open.)
32 (Conditional catalog)	Puts the version in the DSN Catalog with conditional catalog status.
48 (Concatenate datasets)	Reads multiple files using only one open. The files are read as a single dataset. This process is called dataset concatenation. Use concatenation to read multiple files when your program performs only one open. This option is valid for controlled datasets only. For coding, examples, and limitations, see 6-4 in this manual.
64 (Inhibit LUB allocation)	Inhibits LUB allocation for the dataset. Inhibiting LUB allocation is often necessary when you are running programs that use physical IOCS, or the BIM-EPIC TSIDASS program. You do not have to specify this option when LUB allocation is turned off automatically or when a SYS number is specified in the dataset definition.

DLBL Statements

Description

The BIM-EPIC DLBL statement is identical to the VSE DLBL statement, with additional options provided for special BIM-EPIC processing.

BIM-EPIC DLBL options are specified within the file-id (dataset-name) parameter, beginning after the dataset name itself and ending with the closing apostrophe ('). Options must be separated from the dataset name, and from each other, by a comma. The options(s) specified must be enclosed in parentheses.

BIM-EPIC replaces the options, commas, and parentheses with blanks before creating label information, so that only the dataset name appears in VTOC entries.

The maximum length allowed for the dataset name, including options, parentheses, and commas, is 44 characters.

Option Codes

Code	Action
A	Align allocation on cylinder boundaries
B=n	Reblock (n=1 to 32767 MAX OPT)
C	Conditionally catalog
D	Delete after closed as input
E	Concatenate datasets
F	Allow fragmentation
G=n	Access the specified generation (n=1 to 65535)
H	Prevent automatic fragmentation
I	Ignore this dataset
L	Inhibit LUB allocation
M	Issue messages open/close
N	Bypass truncation
O	Use extent size in EXTENT statement
P	Delete (Purge) at End of Job (non-generation files only)
R=n	Allocate by Logical Records (n=1 to 999999)
S	Reuse same space
T	Dataset drop
V=n	Access the specified version (n=1 to 65535)
X	Delete at close or EOJ (non-generation files only)

DLBL Options

A	Aligns the disk allocation on cylinder boundaries. Valid for CKD devices only. Allocations for direct access (DA) and ISAM datasets are aligned automatically.
B=xxx	Changes the block size of your datasets without recompiling your programs. Valid for datasets with either fixed or variable record formats. Reblock with caution when running programs that use physical IOCS; such programs must be able to support reblocking. Specify a numeric block size, MAX for full-track blocking, or OPT for half-track blocking.
C	Puts the version in the DSN Catalog in conditional catalog status.
D	Deletes the dataset after it is closed as input. It is not required for compiler work areas or SORTWK files (they are deleted automatically).
E	Invokes dataset concatenation, which causes the problem program to read two or more files as if they were a single file (using just one open and one close). Dataset concatenation can only be performed for cataloged datasets. Dataset concatenation differs from dataset drop (option 'T') in that dataset drop only processes one dataset when a single open is issued by the program.
F	When the requested amount of space is not available, this option allows BIM-EPIC to allocate the largest available contiguous space. Do not use for TYPEFLE=WORK files (for example, compiler work areas and FORTRAN disk files) or for any other files that do not support secondary extents. If you do not specify the F option, and BIM-EPIC cannot find the requested contiguous extent in the pool, message EP038 is issued, to which operator can respond <i>FRAG</i> to invoke fragmentation.
G=n	Accesses the specified generation of the dataset. This is useful when you want to access a specific version of a dataset without reference to its relative version number. The generation number of a version is listed in the TSIDUTL DSPLY and catalog maintenance DISPLAY reports. It is valid for cataloged datasets only.
H	Overrides the configuration option FRAG=YES, preventing automatic file fragmentation if the space requested for allocation is not available. The EP038 message is issued, so FRAG can still be specified in reply to continue the allocation.

I	Overrides BIM-EPIC's control of dataset. Valid ASSGN and EXTENT statements must be included to process the dataset.
L	Inhibits LUB allocation for the dataset. Inhibiting LUB allocation is often necessary when you are running programs that use physical IOCS. It is required with the BIM-EPIC TSIDASS utility. See "LUB allocation" for more information.
M	<p>Overrides configuration options ALLMALL=NO (default) and ALLMSG=NO for this dataset. Allocation messages are issued for the dataset and for all other disk datasets opened after it.</p> <p>For output datasets, EP039 is issued after the dataset has been opened, displaying the extents allocated. EP052 is issued after the file has been closed, displaying the size of the extent after truncation.</p> <p>For input datasets, EP053 is issued to display the extent that has been opened, and EP051 is issued to indicate that the dataset has been closed.</p> <p><i>This option is ignored for system datasets (such as IJSYSLN) held open across job step boundaries.</i></p>
N	All unused space between the end-of-file record and the end of the last extent remains allocated to the dataset. Use this option when additional records may be written to a file after it is closed if the file is accessed later as a DA file. This option is required for any dataset processed with a TYPEFLE=WORK DTF. This option is also required when you use the 'S' option code (reuse same space).
O	Accepts the size specified in the EXTENT statement as the primary extent rather than the size specified in the DSN Catalog. Pool name and secondary extent size are still taken from the BIM-EPIC dataset definition. Use this option when the volume of data you are processing for a dataset is significantly different from normal. This option is valid for cataloged datasets only.
P	Deletes the dataset at end of job. Use this option for temporary datasets that are not closed as input. This option is automatically activated for all SORTWK files. This option affects only cataloged, non-generation datasets. Configuration option JOBACCT=YES is required.
R=nnn	Specifies the number of logical records the file is to contain. BIM-EPIC uses this number to calculate the size of the extent required.

S	The next version of the dataset is to be written in the same extent space the current version occupies. This option is useful when you want to create datasets under a fixed head for better performance. This option affects single extent non-generation datasets only. When you use this option, you must also inhibit truncation (option 'N') for the dataset.
<hr/>	
T	Deletes the label information for this dataset from the label area (not from the VTOC) when the dataset is closed. If the problem program subsequently issues another open for the file, this label will not be found. This option is useful for programs that issue more than one open for the same DTF name. Dataset drop differs from dataset concatenation (option 'E') in that concatenation causes two or more datasets to be processed as a single dataset when a single open is issued by the program.
<hr/>	
V=n	Specifies a specific version to access. This option is valid for cataloged datasets only. Remember that the latest version is always version 1.
<hr/>	
X	Deletes the dataset at close if it is closed as input or at end of job if it still exists. This option provides the same function as the combination of the 'P' and the 'D' options. The configuration option JOBACCT=YES is required for this option to work. This option is valid only for cataloged, non-generation datasets.
<hr/>	

Example

```
// DLBL INPUT  'INVENTORY,(C,F,E)'
```

6

BIM-EPIC DD Statements

The BIM-EPIC DD statement is an MVS-style dataset label for VSE installations. For many users, its primary benefit is compatibility with MVS, because it simplifies moving JCL between the two operating systems. The DD statement also offers several options which are unavailable with standard VSE labels, even when using BIM-EPIC TLBL and DLBL options.

Chapter Contents

Specifying the DD Statement	6-2
DD Statements for Tape and Disk Datasets	6-7
DD Statements For Printer and Punch Datasets	6-38
DD Statements for In-Stream Datasets	6-39
DD Statements for Dummy Datasets	6-41

Specifying the DD Statement

Introduction

DD Statement Identifier (* //)

All DD statements follow the same format. The identifying characters are '* //', which must be specified in columns 1-4. These are followed immediately by the DD name.

Format

```
* //ddname DD parameter-expressions
```

DD Name

ddname performs the same function as the DTF name (*filename*). It specifies the program's logical name for the dataset. It must begin in column 5.

If you use a DD statement for a DTF that your program defines as a unit record DTF (DTFCD or DTFPR), *ddname* is the SYS number that is specified in that DTF. A unit record DTF is generated for COBOL programs when the SELECT statement contains an assignment name that specifies the class **UR**.

Do not use *ddname* in continuation statements or in the second and following statements of a concatenation.

Some DTF names are invalid for DD statements. Some have special meanings. See 6-5 in this manual for more information.

DD (operation)

DD specifies the DD operation. Place at least one blank immediately before and after it. Do not repeat the DD operation in continuation statements.

Additional Parameters

You can specify many MVS-compatible parameters to define your datasets. The parameters that follow the DD operation specify the type of processing to be done for the dataset. Some parameters are different for:

- Tape and disk datasets (including GDIUR datasets)
- Standard system datasets
- SYSIPT data
- Dummy datasets

Not all MVS DD parameters are supported by the BIM-EPIC DD facility. BIM-EPIC ignores unsupported and unrecognized parameters and subparameters. Supported parameters are discussed in the following sections.

DD Statement Syntax

Specifying Parentheses and Commas

You can specify multiple parameters after the DD operation. Some parameters have multiple subparameters. Separate multiple parameters and subparameters with commas. Enclose multiple subparameters in parentheses.

Many subparameters are positional. If you omit a subparameter, a comma must be specified in the omitted position for all but the last subparameter, just as in VSE and MVS.

Examples

```
* //SYS007 DD DSN=SORTWK,UNIT=SYSDA,DISP=(NEW,DELETE),
* //          SPACE=(TRK,(5,5),RLSE),VOL=SER=DOSRES

VOL=(,RETAIN,,,REF=*.TAPEOUT)
```

Specifying Continuations

All DD statements and continuations can extend through column 71.

Format

```
* //ddname DD parameter,...,parameter,
* //          parameter,...,parameter
```

Step 1

Interrupt the current statement before column 71. The break must occur on a keyword boundary. Do not split subparameters. End the statement with a comma. Do not specify a comment after the comma.

Step 2

Begin the next statement with the statement identifier `* //` in columns 1-4. Omit `ddname` and the DD operation code.

MVS Compatibility

In MVS, the first parameter in a continuation line must begin in or before column 16. If you are specifying a BIM-EPIC DD statement for conversion to MVS, begin your VSE continuations in or before column 18 to maintain compatibility. To move the statement to MVS, delete the asterisk and the space that follows it (`*`), shifting the rest of the statement two columns to the left.

Example

```
* //SYS007 DD DSN=SORTWK,DISP=(NEW,DELETE),
* //          UNIT=SYSDA,VOL=SER=DOSRES,
* //          SPACE=(TRK,(5,5),RLSE)
```

Specifying Dataset Drop

To take advantage of BIM-EPIC's dataset drop option, use multiple DD statements with the same `ddname`. No special coding is required. *This is not valid under MVS. In addition, VSAM datasets (including VSAM-managed SAM datasets) are not eligible for dataset drop.*

Do not insert any other JCL, including `* comment` statements between the DD statements that are part of the "Dataset Drop" set.

Example

```
* //TAPEIN DD DSN=INVENTORY.1QTR(0),DISP=(OLD,KEEP)
* //TAPEIN DD DSN=INVENTORY.2QTR(0),DISP=(OLD,KEEP)
* //TAPEIN DD DSN=INVENTORY.3QTR(0),DISP=(OLD,KEEP)
```

Concatenating Datasets

BIM-EPIC concatenates all versions of a single input dataset automatically, as does MVS. If you do not specify a single version on input, all versions are read sequentially, beginning with the current version.

Example

```
* //TAPEIN DD DSN=INVENTORY.1QTR,DISP=(OLD,KEEP)
```

To concatenate several versions of the same dataset or several different datasets, use a DD statement for each version or dataset and omit the ddname on all but the first statement. Submit the statements in the order you want the datasets processed. If block sizes differ, the version with the largest BLKSIZE *must* be specified first. *VSAM datasets (including VSAM-managed SAM datasets) are not eligible for concatenation.*

Do not insert any other JCL, including * comment statements between the DD statements that are part of the "Concatenating Datasets" set.

Example

```
* //TAPEIN DD DSN=INVENTORY.1QTR(0),DISP=(OLD,KEEP)
* //      DD DSN=INVENTORY.2QTR(0),DISP=(OLD,KEEP)
* //      DD DSN=INVENTORY.3QTR(0),DISP=(OLD,KEEP)
```

Syntax Checking

You can have BIM-EPIC check the syntax of DD statements before you run the job. Place a TYPRUN=SCAN or TYPRUN=SCANS parameter on your JOB statement. See TYPRUN for details.

Special DD Names

Description

Many ddnames have special purposes in MVS. This affects how the BIM-EPIC DD facility processes them.

Unsupported DD Names

Statements containing unsupported ddnames are ignored by the BIM-EPIC DD system and treated as comments by VSE. Use standard VSE JCL to define datasets processed with these names.

The following ddnames are not supported:

- JOBLIB
- STEPLIB
- SYSCHK
- SYSCKEOV
- SYSOUT
- SYSPRINT (changed automatically to SYSOUT)

DD Names with Special Meanings

The following ddnames have special meaning to BIM-EPIC. If a program's DTF uses one of these names for a different purpose, use standard VSE JCL to define the dataset.

- JOBCAT
- STEPCAT
- SYSABEND
- SYSMDUMP
- SYSUDUMP
- SYSIN

JOBCAT and STEPCAT

JOBCAT and STEPCAT define VSAM catalogs to be used for VSAM clusters. In the BIM-EPIC DD facility (but not in MVS), they also define catalogs to be used for VSAM/SAM datasets.

JOBCAT defines the catalog for the entire job stream. STEPCAT defines the catalog for a step. STEPCAT overrides a JOBCAT definition for that step.

A JOBCAT DD must immediately follow the JOB statement. A STEPCAT DD must precede all VSAM labels which require that catalog.

Example

```
//JOB  VSAM UPDATE
* //JOBCAT DD  DSN=VSAM.JOB.CATALOG,DISP=SHR
.

* //STEPCAT DD  DSN=VSAM.STEP.CATALOG,DISP=SHR
vsam labels
// EXEC PROGRAM
/&
```

**SYSABEND,
SYSUDUMP, and
SYSMDUMP**

SYSABEND, SYSUDUMP, and SYSMDUMP determine the kind of dump to be generated when a job cancels. They are similar to the DUMP and PARTDUMP parameters of the VSE OPTION statement. SYSABEND and SYSUDUMP specify a standard partition dump. SYSMDUMP specifies a full supervisor and partition dump.

The SYSOUT parameter must be specified with the SYSABEND, SYSUDUMP, and SYSMDUMP ddnames. SYSOUT identifies a print class in MVS. In VSE, class A indicates SYSLST, class B indicates SYSPCH.

Example

The following statement calls for a standard partition dump. '// OPTION PARTDUMP' is the VSE JCL equivalent.

```
* //SYSABEND DD SYSOUT=A
```

The following statement calls for a full supervisor and partition dump. '// OPTION DUMP' is the VSE JCL equivalent.

```
* //SYSMDUMP DD SYSOUT=A
```

SYSIN

SYSIN is used to assign a logical unit to SYSIPT data. For more information, see 6-39 in this manual.

DD Statements for Tape and Disk Datasets

Introduction to the Parameters

In this section, we present a brief guide to the parameters and their purposes. Following this section, each parameter is examined in detail in a section of its own. Those sections are arranged in alphabetical order for quick reference.

The parameters available for defining tape and disk datasets are:

```
* //ddname DD DSN=dataset-name
           [ ,AMP=x ]
           [ ,DCB=x ]
           [ ,DISP=x ]
           [ ,FREE ]
           [ ,LABEL=x ]
           [ ,OPTION=x ]
           [ ,SPACE=x ]
           [ ,UNIT=x ]
           [ ,VOL=x ]
```

AMP

The AMP parameter is used to define a VSAM cluster or a VSAM/SAM dataset.

DCB

The DCB parameter defines the dataset's record characteristics. DCB is an MVS term that stands for **dataset control block**. It determines the block size, logical record length, density (for tape), and dataset organization (for example, sequential, direct access, or indexed sequential for disk datasets).

DISP

Use the DISP parameter for almost all datasets. It specifies the current status of the dataset and its disposition at normal or abnormal termination.

The DISP parameter specifies whether a dataset is new or already exists, whether it should be retained at close or deleted, whether it should be cataloged or left uncontrolled, and so on.

Because the DISP parameter says so much about the dataset being processed, it also serves as documentation for your JCL, and it is sometimes used explicitly for that purpose even when only the defaults are specified.

DSN

DSN defines the name of the dataset to be processed. It is required for all tape and disk datasets. It also defines which version or versions of the dataset are processed. If you do not specify a particular version, all versions are read in on input (automatic concatenation).

You can also request CPU and partition independence or common work dataset support with the DSN parameter (these options are not available with MVS).

FREE

The FREE parameter specifies how tape drive assignments are to be handled when the dataset is closed. When doing multi-dataset processing, use FREE to hold the assignment until all datasets have been processed. FREE is only valid for tape datasets.

LABEL

The LABEL parameter determines how long output datasets are retained. In most cases, you will use it for uncataloged datasets. You can use it for a cataloged dataset when you wish to override the default retention for a version.

OPTION

OPTION is a BIM-EPIC DD parameter that has no counterpart in MVS. It allows you to specify options available in VSE and BIM-EPIC which have no equivalents in MVS.

SPACE

The SPACE parameter specifies disk space requirements for output datasets. Among other things, it specifies how much space to use and how to process space after the dataset is closed.

SPACE is optional for cataloged datasets. You can use it to specify an allocation size that is different from the size in the DSN Catalog, or to inhibit truncation for a version.

SPACE is required for Start Track 1 and uncontrolled datasets.

UNIT

UNIT specifies the type of device (tape or disk, 3380 or 3390, and so on) to be used for an output dataset.

VOLUME

The VOLUME (VOL) parameter specifies which tape volume, disk volume, or disk pool to allocate for an output dataset. For tape datasets, it can be used to specify rewind options for multi-dataset processing.

What to do for Different Types of Datasets

This section describes the particular DD parameters to use when you are processing different kinds of datasets.

Tape

For tape processing, pay particular attention to the VOLUME, FREE, and DISP parameters. VOLUME has several uses for tape. FREE holds and releases tape drive assignments. DISP=(x,KEEP) unloads tapes at close.

By default a tape is rewound at close, but left mounted. If you specify DISP=(x,KEEP), VOL=PRIVATE, or VOL=(,RETAIN) the rewind default is overridden.

For multi-dataset requirements, see "Multi-Dataset Processing,".

Disk

The SPACE parameter plays a particularly important role in disk processing. Use it to determine the extent allocations for datasets not yet entered in the DSN Catalog. SPACE is always required for Start Track 1 and uncontrolled datasets. Reuse the same extent with DISP=MOD. UNIT is required for disk output.

Cataloged

If the dataset is already defined in the catalog, specify DISP=(x,KEEP) for both input and output. You can override dataset defaults for the version by specifying the equivalent parameter. For instance, you can override period retention by specifying LABEL=RETPD=n. Otherwise, parameters do not have to be specified if that information is already in the dataset definition. If you want to modify the current dataset definition, specify DISP=(x,CATLG). You must also set the JCLOVRDE configuration option to YES.

To autocatalog a new output dataset, specify DISP=(NEW,CATLG). It is cataloged using the data in the DD statement. DD statement parameters cover most of the dataset definition, so you can do a thorough job of defining the dataset in your JCL. You cannot use DD statements to autocatalog an input dataset.

Please note: If a dataset qualifies for autocataloging according to the configuration options you have specified, it will be cataloged automatically even if you don't specify DISP=(x,CATLG). See "Automatic Cataloging" in Chapter 1 for more information on autocataloging with configuration options. The recommended method is to use the DISP=(x,CATLG) parameter to autocatalog all datasets defined with DD statements.

Example

The following example shows a new, cataloged dataset. Its default block size is overridden: its actual block size is 4080.

```
* //TAPEOUT DD DSN=SARS.BACKUP,DISP=(NEW,KEEP),
* //          DCB=BLKSIZE=4080
```

The following example shows a new, uncataloged dataset. It is automatically cataloged, and the specified values for block size, period retention and density are entered in the dataset definition.

```
* //TAPEOUT DD DSN=SARS.BACKUP,DISP=(NEW,CATLG),
* //          DCB=(BLKSIZE=4080,DEN=3),LABEL=RETPD=7
```

Start Track 1 Datasets

To create a Start Track 1 dataset, use the SPACE parameter and specify a relative starting address (SPACE=TRK, SPACE=CYL, or SPACE=alr). You must also use the VOLUME parameter, to assign the dataset to a disk pool. Do not catalog the dataset or specify DISP=(x,CATLG).

Example

```
* //DISKOUT DD DSN=TWOR.BG,DISP=(NEW,PASS),
* //          SPACE=(TRK,(10,5),RLSE),VOL=SER=POOL01,UNIT=DISK
```

Uncontrolled Datasets

To create an uncontrolled (non-VSAM) disk dataset, use the SPACE=ABSTR and VOL parameters. No special parameters are required to create an uncontrolled tape dataset. Do not specify DISP=(x,CATLG), or the dataset will be cataloged and therefore controlled.

Configuration Options

To use BIM-EPIC DD statements for uncontrolled disk datasets, UNCDISK=YES must be specified.

Example

```
* //OPTOUT DD DSN=EPIC.LIBRARY,SPACE=(ABSTR,(100,20)),
* //          DISP=NEW,VOL=SER=SYSWK1,UNIT=DISK
```

Sequential and Direct Access Datasets

See the corresponding topics in "Types of Datasets".

VSAM/SAM

To process a VSAM/SAM dataset, use the AMP, DISP, and SPACE parameters for both input and output. You can use the VOLUME parameter to specify a disk volume. Identify the VSAM catalog with a STEPCAT or JOBCAT DD statement.

Example

In the following example, the STEPCAT DD statement defines the catalog for the VSAM/SAM dataset. AMP=AMORG indicates that a VSAM or VSAM/SAM dataset is being defined. DISP=SHR allows the dataset to be accessed by other programs as it is processed. This example uses ALR to allocate space to the dataset: it indicates a record size of 240 bytes and allocates a primary extent large enough for 1000 records and a secondary large enough for 500 records.

```
* //STEPCAT DD DSN=CUSTOMER.CAT,DISP=SHR
* //VSAMSAM DD DSN=CUSTOMER.MASTER,DISP=SHR,AMP=AMORG,
* //          SPACE=(240,(1000,500))
```

VSAM

You can use existing VSAM datasets using BIM-EPIC DD statements. To process an already existing VSAM dataset, include the AMP parameter in your statement. Define the VSAM catalog with a STEPCAT or JOBCAT DD statement.

Example

```
* //JOBCAT DD DSN=CUSTOMER.CAT,DISP=SHR
* //VSAMCLU DD DSN=CUSTOMER.MASTER,DISP=SHR,AMP=AMORG
```

AMP Parameter

Description

Use the AMP parameter to define a VSAM cluster or a VSAM/SAM dataset.

Syntax

AMP=(AMORG [,BUFSP=x])

Subparameters

Subparameter	Required?	Valid	Default
AMORG	Yes	None	None
BUFSP	No	0 to 9999999	None

AMORG specifies that this is a VSAM cluster or VSAM/SAM dataset.

BUFSP specifies the number of bytes to be allocated as buffer space. If omitted, VSAM will determine the size of the buffer.

DCB Parameter

Description

Use the DCB parameter to define the characteristics of your dataset. It overrides the DSN Catalog entry on output, but not on input.

Syntax

```
DCB=( [ BLKSIZE=x ][ ,DEN=x ][ ,DSORG=x ][ ,LRECL=X ] )
```

Subparameters

Subparameter	Required?	Valid	Default
BLKSIZE	No	1 to 32767 or MAX OPT	None
DEN	No	See DEN table (below)	None
DSORG	No	See DSORG table (below)	PS
LRECL	No	1 to 32767	None

BLKSIZE specifies a block size. It is valid for sequential datasets only. MAX requests full track blocking on disk, and for tape - 64K for VSE/ESA 2.5 and later, or 32K for earlier VSE releases. OPT requests half-track blocking on disk, and for tape - 32K for VSE/ESA 2.5 and later, or 16K for earlier VSE releases.

DEN

Specifies a recording density. Valid density codes are:

Code	Density
2	800 bpi
3	1600 bpi
4	6250 bpi
A	8809 streaming short gap
B	8809 start-stop long gap
C	8809 start-stop short gap
D	8809 streaming long gap
E	3480/3490 non-IDRC buffered write mode
F	3480/3490 IDRC buffered write mode
G	3424/9348 1600 bpi buffered write mode
H	3424/9348 1600 bpi unbuffered write mode
I	3424/9348 6250 bpi buffered write mode
J	3424/9348 6250 bpi unbuffered write mode
K	3490E cartridge device
L	3490E IDRC cartridge device
¬ (X'5F')	None specified - use default density

MVS Compatibility

Densities A through L are not recognized by MVS.

DSORG

Specifies the organization of the dataset. It is required for direct access and indexed sequential datasets. Valid values are:

Code	Organization
DA and DAU	Direct access
IS and ISU	Indexed sequential
PS and PSU	Sequential

MVS Compatibility

There are two options for each type of dataset organization. In MVS, the 'U' indicates that the dataset is unmovable, so the same extents are used every time. When 'PSU', 'DAU', or 'ISU' is specified, and the dataset is a disk work dataset, BIM-EPIC allocates the same extents to the dataset that it already occupies. If you are going to convert to MVS, choose your dataset organization carefully.

LRECL

Specifies the logical record length of the dataset. The number specified must be less than or equal to the record length specified in the program's DTF. If it is greater, the LRECL subparameter is ignored.

DISP Parameter

Description

DISP identifies how a dataset is to be opened and what is to be done to the dataset when it is closed. DISP subparameters define the usage, normal termination and abnormal termination disposition of a dataset. If more than one subparameter is used, the subparameter list must be enclosed in parentheses. When a subparameter inside the list is not specified (to use its default), a comma must still be used for all but the last subparameter.

Syntax

```
DISP=[usage-status][,normal-end-status][,abnormal-end-status]
```

Variables

usage-status is optional. If used it must be one of the following:

- NEW specifies that this is a new dataset or a new version of a dataset.
- OLD specifies that this is an existing dataset
- SHR specifies that this is an existing dataset
- MOD specifies that new data is to be added to an existing version. A new version will be created if there is no existing version.

When you use MOD with a disk dataset, BIM-EPIC will reuse the extents used for the last version. For disk datasets, you can only specify MOD for non-generation disk datasets.

When used with tape, MOD causes new data to be written at the end of the dataset, over the existing EOF record. A new EOF will be created.

WARNING!

MOD does not extend an existing disk dataset as it does for tape, or as MVS does. If MOD is specified for an existing disk dataset that is opened as output, the same extent is used but records are written at the beginning of the dataset. Any data already in the dataset is overwritten and lost.

If *usage-status* is not specified, NEW is assumed.

normal-end-status is optional. If used it must be one of the following:

- CATLG
- CONDCAT
- DELETE
- KEEP
- PASS

If *normal-end-status* is not specified, PASS is assumed if *usage-status* is NEW or MOD. KEEP is assumed if *usage-status* is OLD, SHR, or SHARE.

abnormal-end-status is optional. If used it must be one of the following:

- CATLG

- DELETE
- KEEP

If *abnormal-end-status* is used without both *usage-status* and *normal-end status*, *normal-end status* will be used for *abnormal-end-status*.

Normal Termination Subparameters

DELETE

Causes datasets to be scratched automatically after they have been used. Based upon the dataset usage, the scratching may be done at close, end of step (EOS) or at end of job (EOJ). Several factors determine when and if the DELETE disposition will be effective or bypassed. The factors are:

- work vs. generation dataset
- used for input or output
- for disk, if concatenation is being used

In the cases where the DELETE disposition is bypassed, it will function the same as the PASS disposition.

Deleting Datasets Using DDGEN and the DISP Parameter

Description

The DDGEN configuration option controls when a dataset is to be deleted based upon its DISP DD parameter. The table belows shows when deletion occurs based upon the possible DDGEN and DISP combinations. Where there is no entry in the table, no deletion occurs.

For simplicity, the table below describes the basic DISP combinations. For those combinations not explicitly defined in the table, the following rules apply:

- If the *usage-status* is SHR, it will be treated the same as OLD.
- If the *usage-status* is MOD, it will be treated the same as OLD if there is an existing version. If there is no existing version, it will be treated the same as NEW.
- The CONDCAT *normal-end-status* will be treated the same as KEEP.
- The CATLG *normal-end-status* and *abnormal-end-status* will be treated the same as KEEP.

When DISP=	If DDGEN=NO, then...	If DDGEN=YES, then...
NEW	deletes at end-of-job	deletes at end-of-step
NEW,KEEP		
NEW,KEEP,KEEP		
NEW,KEEP,DELETE	deletes at end-of-job only if job abends	deletes at end-of-step only if step abends
NEW,DELETE	deletes at end-of-job	deletes at end-of-step
NEW,DELETE,KEEP	deletes at end-of-job only if good eoj	deletes at end-of-step only if good eoj
NEW,DELETE,DELETE	deletes at end-of-job	deletes at end-of-step
NEW,PASS	deletes at end-of-job	deletes at end-of-job
NEW,PASS,KEEP	deletes at end-of-job only if good eoj	deletes at end-of-job only if good eoj
NEW,PASS,DELETE	deletes at end-of-job	deletes at end-of-job
OLD		
OLD,KEEP	(Note 1)	(Note 1)
OLD,KEEP,KEEP	(Note 1)	(Note 1)
OLD,KEEP,DELETE	deletes at end-of-job only if job abends	deletes at end-of-step only if step abends
OLD,DELETE	deletes at close (Note 2)	deletes at end-of-step
OLD,DELETE,KEEP	deletes at end-of-job only if good eoj	deletes at end-of-step only if good eoj
OLD,DELETE,DELETE	deletes at close (Note 2)	deletes at end-of-step
OLD,PASS		
OLD,PASS,KEEP		
OLD,PASS,DELETE	deletes at end-of-job only if job abends	deletes at end-of-step only if job abends

NOTE 1 DISP=(OLD,KEEP) and DISP=(OLD,KEEP,KEEP) will automatically reset any delete at end-of-step or delete at end-of-job action currently in effect for the the specified dataset

NOTE 2 Normal deletion will occur at file close but, when used in conjunction with the automatic concatenation feature, deletion will occur at end-of-step. If usage is MOD, deletion will occur at end-of-job.

PURBYP Override In cases of abnormal termination, the end-of-step and end-of-job deletions above can be overridden. To do this, set the PURBYP configuration option to YES.

EPS049 Option BIM-EPIC does not delete datasets at end-of-step or end-of-job unless they have been opened. This can be overridden to allow the deletions even though there has not been an open. To do this, set both the DDGEN configuration option and the EPS049 special option to YES.

Additional DISP Settings

For both normal and abnormal terminations, other DISP settings are treated as follows:

Setting	Is Treated Like...
DISP=MOD	DISP=NEW
DISP=SHR	DISP=OLD.
DISP=(xxx,CATLG)	DISP=(xxx,KEEP)
DISP=(xxx,CONDCAT)	
DISP=(,xxx)	defaults to DISP=(NEW,xxx).

KEEP For tape datasets, if KEEP is explicitly specified, the tape will be unloaded when it is closed. If KEEP is not explicitly specified, the tape will not be unloaded at close. As a default, KEEP does not unload the tape. VOL=RETAIN overrides KEEP.

CATLG (1) Catalogs the dataset automatically if no entry for it already exists in the DSN Catalog or (2) updates the dataset definition if one exists only when configuration option JCLOVRDE=YES is used.

CATLG can cause the following fields to be updated:

Tape/Disk	Tape Only	Disk Only
Retention	Tape pool	Disk pool
Block size		CISIZE
Density		Primary extent size Secondary extent size Maximum number of secondary extents

BIM-EPIC uses the information specified in your DD statement to create or update the dataset definition.

By default, tape datasets are always cataloged as generation datasets, and disk datasets as non-generation datasets. The exception to this occurs when LABEL=RETPD=xx is specified for a

disk dataset; in that case the disk dataset will be cataloged as a generation dataset. If the startup option DDGEN is set to YES, tape and disk datasets will both be automatically cataloged as non-generation datasets unless a plus sign (+) is found in the DSN parameter. See the DDGEN configuration option in the *BIM-EPIC Installation and System Guide* for more information.

Note: The automatic cataloging configuration options (AUTOCAT and AUTOLOG) do not affect CATLG. However CATLG's updating function is permitted by the JCLOVRDE configuration option. Cataloging datasets as generation or non-generation is affected by the DDGEN configuration option.

CONDCAT conditionally catalogs the version. This is not a valid subparameter in MVS.

PASS retains the dataset for use in a later job step.

Abnormal Termination Subparameters

Description

The default disposition for abnormal termination depends on both the normal termination subparameter and the PURBYP configuration option.

KEEP and **CATLG** Retain the dataset if the job cancels.

The abnormal termination disposition is **KEEP** unless both of the following are true:

- Normal termination disposition is **DELETE**
- **PURBYP** is set to **NO**

DSN Parameter

Description

The DSN parameter specifies the name of the dataset being processed. It is required for all tape and disk datasets. It also determines which version or versions of the dataset will be used for input.

The rules for dataset names with DD statements are slightly different from those with standard VSE JCL. Both tape and disk dataset names can be up to 44 characters long. Generation dataset names are limited to 35 characters. If the dataset name contains blanks, enclose it in apostrophes.

Only the first 17 characters of the dataset name are written on a tape's HDR1 label.

Syntax

DSN=dataset-name

.Gnnnn[Vnnn]

(0)

(-n)

(+n)

(area)

or

DSN=&[&]dsname

Subparameters

Subparameter	Required?	Valid	Default
dataset-name	Yes	1 to 44 characters	None
.Gnnnn[Vnnn]	No	Generation number	None
n	No	Zero or signed integer	See text
area	No	INDEX PRIME OVFLOW	None

dataset-name specifies the dataset name.

.Gnnnn[Vnnn] specifies a disk version by generation number. Specify the generation number after the G, but before the V, if any. The Vnnn portion of the parameter is ignored by BIM-EPIC and can be omitted. It is included for MVS compatibility only.

(0), (-n), or (+n) specify a version by version number. (0) indicates the current version. A minus sign (-) indicates a previous version for input. A plus sign (+) indicates a new version for output.

WARNING!

For input, if no version is specified in the dataset name, all versions of a generation dataset will be concatenated and read!

For compatibility with MVS, the current version is indicated by a zero instead of a one as in the DSN Catalog. That means that when you enter a version number with a minus sign, it should be one less than the version number displayed in the DSN Catalog. For example, if the version number in the DSN Catalog is 3, specify `DSN=dataset-name(-2)`.

Also note that specifying `'(+n)'` is not necessary for output datasets. Specifying the dataset name is sufficient to cause a new version to be created. However, specifying `'(+n)'` with `DISP=(x,CATLG)` causes disk datasets to be cataloged as generation datasets, if the DDGEN configuration option is set to YES. If DDGEN is set to NO, and you do not specify `LABEL=RETPD` or `LABEL=EXPDT`, the dataset is cataloged as a non-generation dataset.

(area)

Specifies which area of an ISAM dataset is being defined. INDEX indicates the cylinder index component. PRIME indicates the prime data area. OVFLOW indicates the overflow area.

**&&dsname,
&dsname**

Can be used to specify work datasets. In MVS, placing two ampersands (&&) before a dataset name indicates a temporary dataset. In BIM-EPIC, two ampersands (&&) indicate CPU and partition independence. A single ampersand (&) indicates CPU independence only. The dataset name can be up to 44 characters.

CPU and partition independence can also be indicated with equal signs (=) or dollar signs (\$), as with TLBL and DLBL statements (see "CPU and Partition Independence" and "Common Work Datasets"). Please note, however, that equal signs and dollar signs are not supported by MVS. The MULTCPU configuration option affects CPU independence.

FREE Parameter

Specifies when a tape drive assignment is to be released. FREE is valid for tape datasets only. If FREE is not specified, the assignment is released when the dataset is closed.

When you are processing multi-dataset tapes, you must specify FREE=END for every dataset except the last one processed in the job.

Syntax

FREE={ $\frac{\text{CLOSE}}{\text{END}}$ }

Values

CLOSE releases the assignment when the dataset is closed. This is the default.

END holds the assignment beyond the close of the dataset. The assignment is released at the end of the job.

Example

The following example shows how the FREE parameter is used to hold the assignment until the last dataset is closed.

```
* //INPUT1 DD DSN= ' INPUT.DSN.1 '
* //
* //          UNIT=TAPE,
* //          DISP=(OLD,KEEP),
* //          FREE=END,
* //          VOL=(,RETAIN)
* //INPUT2 DD DSN= ' INPUT.DSN.2 '
* //
* //          UNIT=TAPE,
* //          DISP=(OLD,KEEP),
* //          FREE=END,
* //          VOL=(,RETAIN)
* //INPUT3 DD DSN= ' INPUT.DSN.3 ',
* //
* //          UNIT=TAPE,
* //          DISP=(OLD,KEEP)
```

LABEL Parameter

Description

The LABEL parameter can be used to define the dataset's retention. You can specify an explicit expiration date or specify the number of days the dataset is to be retained.

The LABEL parameter overrides the CYCL and RETEN configuration options. If a dataset is being autocataloged for the first time, the new EDD will take the LABEL value. If a dataset is already cataloged, a new version takes the LABEL value instead of the EDD value.

LABEL is valid only with DISP=NEW. If you specify LABEL=EXPDT or LABEL=RETPD with DISP=(NEW,CATLG), the dataset is cataloged as a generation dataset.

Syntax

```
LABEL={ EXPDT={yyddd or yyyy/ddd}
        RETPD=nnnn }
```

Subparameters

Subparameter	Required?	Valid	Default
Specify only one of the following (required):			
EXPDT		00001 to 99365 2000/001 to 2099/365	None
RETPD		1 to 9999	See below

EXPDT specifies an expiration date for the dataset. Enter the date in Julian format (for example, 92365 for December 31, 1992). If the date is January 1, 2000 or later, you can specify the full year with a slash between the year and the day (for example, 2000/001), or you can omit the slash (2000001).

RETPD specifies the number of days to retain the dataset. If the LABEL parameter is omitted or LABEL=RETPD=7 is specified, the RETEN configuration option value will be used. Otherwise, the value specified will be used.

Examples

```
* //OUTPUT DD DSN=OUT.DATA.SET,LABEL=EXPDT=98365,
* //          DISP=(NEW,KEEP)

* //OUTPUT DD DSN=OUT.DATA.SET,LABEL=RETPD=5,
* //          DISP=(NEW,KEEP)
```

OPTION Parameter

Description

The OPTION parameter can be used to invoke a BIM-EPIC feature applicable for which there is no equivalent MVS JCL parameter. OPTION can be used to:

- selectively activate disk allocation messages
- specify the name of an applicable VSAM catalog
- specify an FBA CFSIZE
- specify a specific logical unit number
- specify a tape pool ID
- base current disk allocation size on the size used by a different dataset
- selectively remove a VSE label from the label area
- override rewind options specified by DISP or VOL parameters

When multiple subparameters are used, they must be separated by commas and the list must be enclosed in parentheses. The subparameters are not positional, so it is not necessary to use commas in place of unspecified subparameters.

Syntax

$$\text{OPTION}=\left\{ \begin{array}{l} [,\text{ALLMSGs}] \\ [,\text{CAT}=\text{ddname}] \\ [,\text{CFSIZE}=\text{n}] \\ [,\text{NOREW}] \\ [,\text{RESET}] \\ [,\text{REW}] \\ [,\text{RUN}] \\ [,\text{SPACE}=\text{*}.\text{ddname}[\text{x}]] \\ [,\text{SYSnnn}] \\ [,\text{TPL}=\text{x}] \end{array} \right\}$$

Subparameters

Subparameter	Required?	Valid	Default
ALLMSGs	No		OFF
CFSIZE=	No	512, 1024, 2048, 3072, 4096	None
NOREW	No		see below
RESET	No		OFF
REW	No		see below
RUN	No		see below
SPACE=	No	1 to 15 characters	None
nnn	No	000 to 254 RDR IPT LST PCH	None
TPL=	No	A to Z or 0 to 9 (Use "-" {X'5F'} to indicate no TPL value is desired)	None

ALLMSG	Overrides configuration options ALLMALL=NO (default) and ALLMSG=NO for this dataset in the current job step only. It enables all allocation messages for the disk dataset. This subparameter performs the same function as the BIM-EPIC DLBL option 'M'. The ALLMSG subparameter is valid for disk datasets only and is ignored if specified for a tape dataset.
CAT=ddname	Specifies the DD name used in a previous DD or DLBL statement to define this dataset's VSAM catalog. Use this parameter only when you are defining a VSAM dataset. It will override any JOBCAT or STEPCAT statement for the dataset.
CISIZE=nnnn	Specifies the CISIZE to be used for an FBA disk dataset according to IBM rules. NOTE: If the dataset is a Start Track 1 dataset and CISIZE= is specified, specification of the maximum number of secondary extents in the VOLUME parameter is ignored.
NOREW	OPTION=NOREW can be used to not rewind the tape at close. It overrides any rewind option defaulted to by the DISP or VOL parameters.
RESET	<p>Specifies that the control of a dataset by the Generic Device Independence for Unit Record (GDIUR) feature is to be ended. If OPTION=RESET is used, it must be the only parameter specified on the DD statement, and no other OPTION subparameter can be used.</p> <p>A DD statement with OPTION=RESET is only required when the job contains no TLBL, DLBL, or DD statements in any step following the DD statement that was used to invoke GDIUR control of the dataset. The ddname on the DD statement that specifies OPTION=RESET must be the same as the ddname on the DD statement that was used to invoke GDIUR control of the dataset.</p>
RESET Example	<p>In this example, SYSIPT control statements are read by the first LIBR execution from the tape or disk dataset "SYSIPT.DATA". The second DD statement for IJSYSIN with the OPTION=RESET insures that the SYSIPT control statements for the second LIBR execution will be read from the instream data, not from the "SYSIPT.DATA" dataset.</p> <pre> * //IJSYSIN DD DSN=SYSIPT.DATA,DISP=OLD // EXEC LIBR /* * //IJSYSIN DD OPTION=RESET // EXEC LIBR LD L=IJSYSRS /* </pre>
REW	OPTION=REW can be used to rewind the tape at close. It overrides any rewind option defaulted to by the DISP or VOL parameters.

RUN

OPTION=RUN can be used to unload the tape at close. It overrides any rewind option defaulted to by the DISP or VOL parameters.

**SPACE=
*.ddname[x]**

Specifies that the primary extent allocation is to be based on the amount of space used by a dataset that was defined previously in the same job step. Identify the dataset you are referring to by specifying its ddname in the SPACE parameter. RLSE is automatically invoked with SPACE refer-back.

There are several ways in which the size of the allocation can be varied:

1. To specify that the dataset is to be exactly the same size as the referenced dataset, use the format:

```
SPACE=* .ddname
```

2. To specify that a dataset is to be larger than the referenced dataset by an exact number of tracks or FBA blocks, use the format:

```
SPACE=* .ddname+n
```

where n is the number of tracks or FBA blocks to be added to the size of the referenced dataset.

3. To specify that a dataset is to be smaller than the referenced dataset by an exact number of tracks or FBA blocks, use the format:

```
SPACE=* .ddname-n
```

where n is the number of tracks or FBA blocks to be subtracted from the size of the referenced dataset. Note that an error will result if the calculated allocation is not a positive number.

4. To specify that the size of the dataset is to be a percentage of the size of the referenced dataset, use the format:

```
SPACE=* .ddname*n
```

where n is a number between 000.01 and 999.99 (leading zeroes are not required).

If the referenced dataset is cataloged, the allocation will be based on the amount of space the referenced dataset actually occupies (the sum of its primary extent space and all secondary extent spaces). If the referenced dataset is uncataloged, the allocation will be based on the amount of space in the referenced dataset's primary extent only.

SPACE Examples

The following DD statement defines the referenced dataset:

```
* //REFFILE DD DSN=dataset-name,DISP=SHR
```

To allocate a new dataset one track larger than the space the referenced dataset occupies, specify:

```
* //FILEOUT DD DSN=dataset-name,DISP=(NEW,CATLG)
* //          UNIT=SYSDA,OPTION=SPACE=* .REFFILE+1
```

To allocate a new dataset five tracks smaller than the space the referenced dataset occupies, specify:

```
* //FILEOUT DD DSN=dataset-name,DISP=(NEW,CATLG)
* //          UNIT=SYSDA,OPTION=SPACE=*.REFFILE-5
```

To allocate a new dataset half of the space occupied by the referenced dataset, specify:

```
* //FILEOUT DD DSN=dataset-name,DISP=(NEW,CATLG)
* //          UNIT=SYSDA,OPTION=SPACE=*.REFFILE*.5
```

To allocate a new dataset twice the space occupied by the referenced dataset, specify:

```
* //FILEOUT DD DSN=dataset-name,DISP=(NEW,CATLG)
* //          UNIT=SYSDA,OPTION=SPACE=*.REFFILE*2
```

SYSnnn

Causes the specified SYS number to be assigned to this dataset. Replace *nnn* with the appropriate programmer logical unit number or 'RDR', 'IPT', 'PCH', or 'LST'.

You can use `OPTION=SYSnnn` to satisfy the pre-open checking that is performed by some programs, such as IBM's SORT/MERGE. For most jobs, this eliminates the need to execute TSIDASS ASSGN for tape datasets or to use `// ASSGN` statements for disk datasets before such programs. For disk datasets, ensure that the pool name is specified in the DSN Catalog or on the `VOL=SER` parameter.

SYSnnn Example

```
// JOB IDCAMS  REPRO DISK TO TAPE
* //IN      DD DSN=DISK.SEQ,DISP=OLD,
* //        UNIT=DISK,OPTION=SYS007
* //OUT     DD DSN=DISK.SEQ.BACKUP,OPTION=SYS005,
* //        DISP=(NEW,KEEP,DELETE)
// EXEC IDCAMS,SIZE=AUTO
  REPRO INFILE(IN -
    ENVIRONMENT (RECFM(FIXBLK) BLKSZ(800) RECSZ(40) )) -
    OUTFILE(OUT -
    ENVIRONMENT -
    ( PDEV(2400) RECFM(FIXBLK) BLKSZ(8000) RECSZ(40) ))
/*
/&
```

TPL=x

Specifies a tape pool for new tape datasets. `TPL=x` is honored only when `DISP=(NEW,CATLG)` is specified causing a new data set to be automatically cataloged or an existing dataset to be updated (only allowed with configuration option `JCLOVRDE=YES`).

SPACE Parameter

Description

SPACE specifies the allocation of a disk dataset. Space is used to:

- specify allocation size
- permit or inhibit truncation at close for an output dataset
- invoke dataset fragmentation if contiguous space is unavailable
- align allocation on track or cylinder boundaries on CKD devices

Parentheses are used to enclose the positional subparameters of the SPACE parameter. Additional parentheses are used to enclose the actual allocation amounts when supplying a prime allocation size with a secondary size and/or an index size. SPACE has two types of syntax, one for BIM-EPIC-controlled datasets and one for uncontrolled datasets.

Configuration Options

Using DD statements for uncontrolled disk datasets is permitted by the UNCDISK configuration option.

Controlled Dataset Syntax

`SPACE=(type,(size-spec),RLSE,frag-control,ROUND)`

type is required. It must be one of the following:

- TRK specifies allocation in tracks or FBA blocks.
- CYL specifies allocation in cylinders and forces the dataset to begin and end on cylinder boundaries.
- *block-length* specifies the dataset's block size for allocation by number of logical records (ALR). *block-length* can also be used to specify the record size of VSAM/SAM datasets.

size-spec is required. It must be enclosed in parentheses and takes the form:

`primary-qty[,secondary-qty][,index]`

where *primary-qty* specifies the amount of the primary allocation in tracks, FBA blocks, cylinders, or number of logical records depending on the value specified by type. To use the SPACE parameter to indicate an option for a cataloged dataset (such as fragmentation, no truncation, or align on cylinder boundaries) without affecting the allocation size, specify zero (0) in *primary-qty*. When *primary-qty* is zero, BIM-EPIC uses the allocation size in the dataset definition.

secondary-qty specifies the size of the secondary allocation. To specify a maximum number of secondary allocations, use the VOLUME parameter.

index specifies the size of the index component for an ISAM dataset. Use it only when both the prime and the index areas are defined with a single set of parameters. If you use *index* but omit *secondary-qty*, be sure to specify the comma for *secondary-qty*.

RLSE is optional. RLSE causes truncation of the dataset at close. The default is no truncation if RLSE is not specified, even for cataloged datasets (when the SPACE parameter is

specified). However, if the SPACE parameter is not specified for cataloged datasets, they are truncated at close.

frag-control is optional. If used, it must be one of the following:

- CONTIG specifies that each allocation must be made using consecutive tracks or blocks for the entire allocation amount. This is the same as the DLBL option 'H'.
- MXIG and ALX both specify fragmentation without operator intervention. This is the same as the DLBL option 'F'.

Specify only one of the CONTIG, MXIG, or ALX subparameters.

MVS Compatibility

MXG and ALX have different meanings in MVS. For more information, see an MVS JCL manual.

ROUND is optional. It specifies alignment on a cylinder boundary.

Controlled Dataset Examples

In the following example, the dataset asks for primary and secondary allocations of 5 cylinders. Unused space will be released. The dataset will be automatically cataloged.

```
* //DISKIN DD DSN=INVENTORY.FILE,SPACE=(CYL,(5,5),RLSE),
* //          DISP=(NEW,CATLG)
```

In the following example, exactly the same space requirements are set, but because the DISP parameter doesn't tell BIM-EPIC to automatically catalog the dataset, it is treated as a Start Track 1 dataset. Note that the VOLUME parameter is also specified because it is required for a Start Track 1 dataset.

```
* //DISKIN DD DSN=INVENTORY.FILE,SPACE=(CYL,(5,5),RLSE),
* //          DISP=(NEW,KEEP),VOL=SER=SYSWK1
```

Uncontrolled Dataset Syntax

SPACE=(ABSTR,(size-specification))

ABSTR is required. It specifies the allocation requirements for an uncontrolled dataset. The dataset will be allocated at the address you specify in the next parameters.

size-specification is required. It must be enclosed within parentheses and takes the form:

primary-qty, start-address [,index]

where *primary-qty* specifies the size of the primary allocation in tracks or FBA blocks.

start-address specifies the starting relative track or FBA block (as in the VSE EXTENT statement).

index is only used when both the prime and index data areas are defined using a single set of parameters. It specifies the size of the index component for an ISAM dataset.

primary-qty, *start-address*, and *index* values are positional and must be enclosed in parentheses.

Example

The following example shows an uncontrolled dataset. BIM-EPIC creates a label indicating the extent begins at relative track 1680 and is 5 tracks long. It is on the SYSWK1 volume.

```
* //DISKIN DD DSN=INVENTORY.FILE,SPACE=(ABSTR,(5,1680)),
* //          DISP=(NEW,KEEP),VOL=SER=SYSWK1
```

UNIT Parameter

Description

Specifies a device or device type for processing. It is usually specified for output, and it is required for disk output datasets. It is optional for input datasets that are controlled by the DSN Catalog. If UNIT is not specified, UNIT=TAPE is assumed unless the dataset name begins with “\$\$\$.” (common work dataset) in which case UNIT=DISK is forced.

You can also request a unit by a group name. For disk datasets, BIM-EPIC chooses a volume from the disk pool specified on the VOLUME parameter. For all tape datasets, BIM-EPIC selects a suitable tape drive based on the mode setting associated with the group name/device-type, if one exists, or associated to the default MODE setting.

For uncontrolled disk datasets, you can use a group name such as 'UNIT=DISK'. However, you must also specify a particular device with the VOLUME parameter.

MVS Compatibility

In MVS, you can specify a pool name in the UNIT parameter. With BIM-EPIC DD statements, you must specify the VOL=SER parameter instead.

Syntax

$$\text{UNIT}=\left\{\begin{array}{l} \text{group-name} \\ \text{device-type} \text{ [,cuu]} \end{array}\right\}$$

Parameters

Parameter	Required?	Valid	Default
Specify one of the following (required):			
<i>group-name</i>		DISK SYSDA TAPE IDRC CART	TAPE
<i>device-type</i>		See text	None

group-name specifies a generic request for a device group. This is the most generic kind of request. In MVS it specifies a particular category of device set up by your installation. In BIM-EPIC valid groups are:

- DISK or SYSDA specify any disk device
- TAPE specifies any tape device (does not override default MODE)
- IDRC specifies an IDRC drive
- CART specifies a 3480 non-IDRC drive
- TPA specifies a 3590 drive

device-type specifies a device type, such as FBA, 3390, 3480. Valid device types are listed in the IBM *System Control Statements* manual. See the tape and disk device-classes for the ASSGN statement. For tape devices, following are the associated mode settings (if not overridden by the DCB DEN parameter):

- 2400 specifies any tape device (does not override default MODE)
- 3410 specifies any tape device (does not override default MODE)

- 3420 specifies any tape device (does not override default MODE)
- 3424 specifies any tape device (does not override default MODE)
- 3430 specifies any tape device (does not override default MODE)
- 3480 specifies a 3480 non-IDRC drive
- 3490 specifies a 3490 IDRC drive
- 3590 specified a TPA device
- 8809 specifies any tape device (does not override default MODE)
- 9348 specifies any tape device (does not override default MODE)

cuu specifies the address of a particular device. If *cuu* is specified, it must be separated from device-type by a comma, and both subparameters must be enclosed in parentheses. For example, UNIT=(CART,3F0).

Example

In the example below, the UNIT parameter indicates that this is a tape dataset. Any tape drive can be used.

```
* //TAPEOUT DD DSN=INVENTORY.FILE,UNIT=TAPE,
* //          DISP=(NEW,CATLG)
```

VOLUME Parameter

Description

For tape datasets, use **VOLUME** to specify tape rewind options and to select a tape volume to process. These options are especially important for multi-dataset processing.

For disk datasets, use **VOLUME** to specify a disk pool for allocation and to specify a maximum number of secondary allocations. **VOLUME** is required for Start Track 1 datasets.

VOLUME's first four subparameters are positional. The last subparameter, **SER** or **REF**, is both keyword and positional. If specified alone, no commas are required before it. If other subparameters are specified as well, **SER** or **REF** must be the last positional subparameter.

Syntax

$$\text{VOL=} \left(\begin{array}{l} [\text{PRIVATE}] \\ [,\text{RETAIN}] \\ [,\text{volume-sequence-number}] \\ [,\text{secondary-max}] \\ [,\text{SER=volser}] \\ [,\text{SER=poolname}] \\ [,\text{REF=*.ddname}] \end{array} \right)$$

Subparameters

Subparameter	Required?	Valid	Default
PRIVATE	No		None
RETAIN	No		None
<i>volume-sequence-number</i>	No	1 to 255	1
<i>secondary-max</i>	No	1 to 255	255
SER= <i>volser</i>	No	1 to 6 characters	None
SER= <i>poolname</i>	For ST1	1 to 6 characters	None
REF=*. <i>ddname</i>	No	1 to 7 characters	None

PRIVATE causes the tape to be unloaded when it is closed. **PRIVATE** is overridden by the **RETAIN** parameter. If **PRIVATE** is not specified, the tape is rewound when it is closed. **PRIVATE** is ignored for disk datasets.

RETAIN prevents rewinding the tape when it is closed. This subparameter is most often used when processing multi-datasets. If neither **PRIVATE** nor **RETAIN** is specified, the tape is rewound when it is closed. **RETAIN** is ignored for disk datasets.

volume-sequence-number selects the first reel to process in a multi-volume group. It is ignored for disk datasets.

secondary-max specifies the maximum number of secondary allocations which can take place for this disk dataset. It is ignored for tape datasets.

SER=*volser* selects a volume by volume serial number. You can use this subparameter to specify a disk volume for uncontrolled disk datasets.

SER=*poolname* selects a disk pool. It is required for Start Track 1 datasets.

Example

In the following example, the VOL parameter indicates the pool where the Start Track 1 dataset is located.

```
* //DISKIN DD DSN=INVENTORY.FILE,SPACE=(CYL,(5,5),RLSE),
* //          DISP=(NEW,KEEP),VOL=SER=SYSWK1
```

REF=*ddname* associates a subdataset with a prime dataset. Use it with DISP=(NEW,CATLG) when you want to automatically catalog a subdataset. You can also use it when you want to refer to a prime dataset other than the one specified in the subdataset's dataset definition. Both the prime dataset and the subdataset must be controlled, and both must be defined with DD statements. The prime dataset's DD statement must precede all DD statements for related subdatasets.

Specify the ddname that is on the prime dataset's DD statement. For example, specify:

```
REF=*.TAPEOUT
```

to refer to a prime dataset defined with the ddname TAPEOUT.

Example

In the following example, the tape is unloaded at close.

```
* //TAPEIN DD DSN=INVENTORY.FILE,
* //          DISP=(NEW,KEEP),VOL=(PRIVATE)
```

The following example automatically catalogs a prime dataset and two new subdatasets. The datasets are "stacked" on one tape to create a multi-dataset group. 'PRIME.DSN.1' is the first dataset on the tape, 'NEWSUB.DSN.2' is the second dataset and 'NEWSUB.DSN.3' is the third dataset.

```
* //TAPEDUM DD DSN='PRIME.DSN.1(+0)',
* //          OPTION=SYS010,
* //          UNIT=TAPE
* //          DISP=(NEW,CATLG),
* //          FREE=END,
* //          VOL=(,RETAIN)
* //TAPEOUT DD DSN='NEWSUB.DSN.2(+0)',
* //          OPTION=SYS010,
* //          UNIT=TAPE,
* //          DISP=(NEW,CATLG),
* //          FREE=END,
* //          VOL=(,RETAIN,,,REF=*.TAPEDUM)
* //TAPEOUT DD DSN='NEWSUB.DSN.3(+0)',
* //          OPTION=SYS010,
* //          UNIT=TAPE,
* //          DISP=(NEW,CATLG),
* //          VOL=(PRIVATE,,,REF=*.TAPEDUM)
// EXEC PROGX
```

The VOL=REF=*subparameter* tells BIM-EPIC the name of the prime dataset.

VOL=(RETAIN,,,REF=*.TAPEDUM) tells BIM-EPIC that both of the following are true:

- this dataset is a subdataset and part of a multi-dataset group
- the prime dataset associated with this subdataset is 'PRIME.DSN.1'

See "Multi-Dataset Processing" for further explanation of prime datasets and subdatasets.

DD Statements For Printer and Punch Datasets

Description

You can use DD statements to assign printer and punch datasets.

Format

```
* //ddname DD SYSOUT=x [ ,OPTION=SYSnnn ]
```

Parameters

Parameter	Required?	Valid	Default
<i>x</i>	Yes	A to Z or 0 to 9	None
<i>SYSnnn</i>	No	SYS000 to SYS254	None

SYSOUT specifies the class to be used for printer and punch data. For punch data, specify B. For print data, specify any other character.

SYSnnn specifies the programmer logical unit. Use this subparameter if the ddname itself is not a logical unit specification.

For example, if your COBOL program includes the statement

```
SELECT PRINT-OUT ASSIGN SYS006-UR-1403-S-PRINT.
```

then the DD statement would be:

```
* //PRINT DD SYSOUT=A,OPTION=SYS006
```

If no external name is used (the ddname is itself a logical unit specification), do not use the OPTION parameter. For example, if your COBOL program includes the statement

```
SELECT PRINT-OUT ASSIGN SYS006-UR-1403-S.
```

then the DD statement would be:

```
* //SYS006 DD SYSOUT=A
```


DD Statements for In-Stream Datasets

Description

When a job uses in-stream data, you can use a DD statement to assign a logical unit to the data stream. You must still place the in-stream data after the EXEC statement, as in VSE.

Format

```
* //ddname DD specifier [,OPTION=SYSnnn]
```

Parameters

Parameter	Required?	Valid	Default
<i>specifier</i>	Yes	* or DATA	None
<i>SYSnnn</i>	No	SYS000 to SYS254	None

* or DATA specifies that this statement defines an instream dataset.

SYSnnn specifies a programmer logical unit. Use this parameter if the ddname itself is not a logical unit specification.

For example, if your COBOL program includes the following statement:

```
SELECT CARD-IN ASSIGN SYS005-UR-2540R-S-CARDIN.
```

then the DD statement would be:

```
* //CARDIN DD *,OPTION=SYS005
or
* //CARDIN DD DATA,OPTION=SYS005
```

If no external name is used (the ddname is itself a logical unit specification), do not use the OPTION parameter.

For example, if your COBOL program includes the following statement:

```
SELECT CARD-IN ASSIGN SYS005-UR-2540R-S.
```

then the DD statement would be:

```
* //SYS005 DD *
or
* //SYS005 DD DATA
```

Examples

In the following example, the AJS00987 statement is read as in-stream data using SYS005. Because the ddname is itself a logical unit specification, no OPTION statement is necessary.

```
* //SYS005 DD *  
// EXEC PGM  
AJS00987  
/*
```

In the following example, LOG JE0987 and SP 9876 are read as in-stream data using logical unit SYS005.

```
* //CARDIN DD DATA,OPTION=SYS005  
// EXEC PGM  
LOG JE0987  
SP 9876  
/*
```

DD Statements for Dummy Datasets

Description

You can define dummy datasets with DD statements.

Format

```
* //ddname DD DUMMY [ ,OPTION=SYSnnn]
```

Parameters

Parameter	Required?	Valid	Default
DUMMY	Yes		None
SYSnnn	No	SYS000 to SYS254	None

DUMMY specifies that this is a dummy dataset.

SYSnnn specifies the programmer logical unit and is required if the ddname itself is not a logical unit specification.

Example

This example defines the dataset belonging to DTF name (and logical unit) SYS005 as a dummy dataset. It is equivalent to a VSE // ASSGN SYS005,IGN. Input and output to the dataset is ignored.

```
* //SYS005 DD DUMMY
```


7

Running Jobs With BIM-EPIC

BIM-EPIC is designed to work with your existing JCL. However, JCL changes are sometimes necessary and sometimes helpful. Some situations require special entries in the BIM-EPIC dataset definition (EDD). This chapter discusses how to use BIM-EPIC and JCL with products from IBM and other third-party vendors. It describes new requirements and the use of BIM-EPIC label options. Recommendations for JCL and EDDs are presented for each topic when relevant. Please refer to the \$INDEX SAMPJOB file as a reference to the many job samples provided on the BIM-EPIC installation tape. Familiarity with the TLBL and DLBL statements and BIM-EPIC DD statements is assumed.

Chapter Contents

Introduction.....	7-2
Disk Processing.....	7-8
Processing Tape Datasets.....	7-11
Types of Datasets.....	7-24
Sorting with BIM-EPIC.....	7-37
IDCAMS BACKUP and RESTORE.....	7-43
IDCAM\S REPRO.....	7-44
IDCAMS IMPORT/EXPORT.....	7-45
LIBR BACKUP/RESTORE.....	7-46
DITTO.....	7-47
VSE/POWER ACCOUNT FILE.....	7-48
VSE/POWER PDISPLAY, POFFLOAD, AND Printer/Punch Tape Spooling.....	7-49
SQL/VSE Tape Output Functions.....	7-51
VSE/ICCF and Interactive Partition GETVIS Requirements with BIM-EPIC.....	7-52
Processing with COBOL II.....	7-53
Using BIM-EPIC's JCL Features.....	7-54

Introduction

Unsupported Datasets

Description

BIM-EPIC does not control datasets that are defined in the JCL with the following DTF names:

- IJSYSNR
- IJSYSPC
- IJSYSRS

BIM-EPIC also does not control datasets processed with the following system utilities:

- DITTO basic and control functions

BIM-EPIC can control DITTO file functions.

JCL or DSN Catalog?

Description

For output datasets, JCL parameters always override corresponding DSN Catalog information. However, JCL period retention must be greater than zero to override a DSN Catalog value.

For input datasets, DSN Catalog values override any corresponding JCL parameters, with the following exceptions:

- EXTENT size (DLBL or DD SPACE parameter)
- density (TLBL)
- block size or CISIZE (DLBL or DD DCB parameter)
- period retention (days) and expiration date

Input Dataset Selection

Description

When you use TLBL and DLBL statements for input, BIM-EPIC selects the current version by default.

When you use DD statements for input, the entire dataset is concatenated if you do not specify a version number. All versions of the dataset are read, beginning with the current version and ending with the oldest version.

JCL can be used to override these defaults. You can request a specific tape version or a specific disk version or generation. For more details on overriding defaults, please see 5-2, 5-8, and 6-1 in this manual.

Defining Your Datasets With TLBL, DLBL, and DD Statements

Description

For every controlled dataset that it processes, BIM-EPIC uses the information in the DSN Catalog entry to update the VSE label area when the dataset is opened. Because this information is recorded automatically, JCL can be very concise. Using the DSN Catalog instead of JCL can be:

- simpler
- less error-prone
- easier to change

In many circumstances, all you need to specify for an input dataset is the DTF name (or the ddname that the program uses) and the dataset name.

Examples

```
// TLBL TAPEOUT, 'SYSTEM.BACKUP'
```

```
* //TAPEOUT DD DSN=SYSTEM.BACKUP
```

On DD statements for output datasets, you may also need to specify a device type.

```
* //DISKOUT DD DSN=SYSTEM.BACKUP,UNIT=DISK
```

Rules For Dataset Name

Description

Rules for dataset names are specified at 5-2, 5-8 and 6-22 in this manual.

For generation datasets, specify the dataset name as it appears in the EDD. *Do not specify the G=nnnnnn extension that appears in the VTOC.*

LUB Allocation and the ASSGN Statement

When an ASSGN Statement is Not Required

Using LUB allocation and AVR, BIM-EPIC does the work of an ASSGN statement. LUB allocation selects the SYS number, and AVR selects the drive. As long as LUB allocation is active, ASSGN statements are not necessary and should be removed from your JCL.

When an ASSGN Statement is Required

ASSGN statements must be used whenever LUB allocation is inhibited. When LUB allocation is inhibited, the drive specified in the ASSGN statement is used.

LUB allocation is automatically inhibited:

- for direct access datasets
- if the logical unit number is currently assigned IGNORE
- for system logical units
- if the logical unit is assigned in the DSN Catalog

and for the following DTF names:

- UIN
- UOUT
- IJSYSxx
- SYS000
- SORTxxx (unless TSIDSRT is executed in place of SORT)

You can add other DTF names to this list by specifying them in the TSIDMAP macro. See the *BIM-EPIC Installation and System Guide* for details.

In addition, you must inhibit LUB allocation for datasets opened by TSIDASS and datasets which will be processed with PIOCS. This includes datasets processed with many utility programs and any dataset opened with a DTFPH. In most circumstances, the TSIDASS utility can replace the use of a VSE ASSGN statement.

Uncontrolled tape datasets do not need ASSGN statements. BIM-EPIC will request the operator to specify to a tape drive address.

Inhibiting LUB Allocation

You can inhibit LUB allocation for a dataset in several ways:

- the "L" option in the dataset name on a DLBL
- option "64" in the numeric options on a TLBL
- OPTION=SYSnnn parameter on a DD statement
- a default SYS number in the BIM-EPIC dataset definition

Disk Processing

Controlled Datasets

EXTENT Statements

For most controlled datasets, the EXTENT statement is ignored. An EXTENT statement can be required when you process direct access (DA) datasets or any datasets with programs that use PIOCS processing. See "Physical IOCS Processing" in this chapter for details.

To change the extent size from the size in the EDD, use the DLBL statement with the "O" option in the dataset name and use the EXTENT statement as you would for VSE. All parameters except the number of tracks/blocks on the EXTENT are ignored.

Example

```
// DLBL DISKIN, '==.WORKFILE.1,(O)'  
// EXTENT SYS004,SYSWK1,1,0,5,210
```

This dataset is created with a primary extent size of 210 tracks, overriding the primary extent size in the EDD.

Start Track 1 Datasets

Description

To define a Start Track 1 dataset, you must specify allocation information.

DLBL Statements

EXTENT statements are required for Start Track 1 datasets. The EXTENT statement for a Start Track 1 dataset differs only slightly from the standard VSE EXTENT statement. To use a Start Track 1 EXTENT, do the following:

- Specify a BIM-EPIC disk pool instead of a volume serial number.
 - Specify a relative starting track/block of 1.
-

Example

A Start Track 1 dataset is allocated in POOL01 with 20 tracks.

```
// EXTENT SYS003,POOL01,1,0,1,20
```

If you specify a Start Track 1 EXTENT for a controlled dataset, BIM-EPIC ignores the EXTENT information. For Start Track 1 datasets defined with BIM-EPIC DD statements, see 6-10 in this manual.

Reusing the Same Space

DSN Catalog

If you want to write a new version of a disk dataset in the same location (reusing the same space), catalog the dataset as work ("WRK" using Online Manager, or WRK=YES using TSIDMNT) and zero secondary allocations.

JCL Requirements

1. Each time the dataset is created or modified, inhibit truncation. This leaves space in the extent for a larger dataset the next time it is processed. To inhibit truncation with a DLBL, specify the "N" option in the dataset name; with a DD statement, use the SPACE parameter.
 2. Each time the dataset is modified, you must indicate that you want to reuse the same space. With a DLBL, specify option "S"; with a DD statement, specify DISP=(MOD,KEEP).
-

Examples

```
// DLBL OUTPUT, 'DATA.FILE', (N,S) '
* //OUTPUT DD DSN=DATA.FILE,UNIT=DISK,
* //          DISP=(MOD,KEEP),SPACE=(TRK,0)
```

Processing Tape Datasets

Processing Internal Tape Labels

Description

BIM-EPIC supports both labeled and unlabeled tape processing. BIM-EPIC automatically changes the DTF for an unlabeled tape (FILABL=NO or "LABEL RECORDS OMITTED") to a DTF for a labeled tape (FILABL=STD or "LABEL RECORDS STANDARD") when you supply a TLBL with the DTF name. This allows you to convert to labeled processing without making program changes. BIM-EPIC does not affect user header label processing.

DSN Catalog

To indicate standard label processing, specify LBL=YES on the TSIDMNT CATALOG statement. To indicate unlabeled processing, specify LBL=NO.

JCL Requirements

When processing a controlled dataset on an unlabeled tape, supply a TLBL or a DD statement for the dataset. Standard VSE processing does not require a TLBL for unlabeled tape datasets.

Rewind Options and MTC Statements

You do not need to use MTC statements to rewind your tapes. BIM-EPIC will position the tape automatically for input processing. You can also use BIM-EPIC's rewind options for output or input processing. These label options will override equivalent options in your program.

See `OPTION` at page 6-26 in this manual, `PRIVATE` and `RETAIN` at page 6-35 in this manual, and also `TLBL` options 1, 2, and 3 at page 5-6 in this manual for information on tape rewind options.

Multi-Volume Processing

Description

You can process up to 255 volumes per tape dataset version and up to 255 extents per disk dataset version.

JCL Requirements

None.

Multi-Dataset Processing

Description

This section explains the concept of **multi-dataset groups** and describes their processing requirements and options.

A multi-dataset group consists of a prime dataset and up to 9,998 subdatasets. A multi-dataset group can reside on a single volume or on multiple volumes.

WARNING!

All datasets participating in a multi-dataset group **MUST** conform to standard dataset format as defined by LIOCS (Logical Input Output Control System). Standard dataset format is comprised of three components, each followed by a tape mark:

- Header label set
- Data
- Trailer label set

Nearly all programming languages and utility programs create datasets which conform to standard format.

Datasets known not to conform to standard data format are those created by the LIBR program BACKUP function and the IDCAMS program BACKUP function. Datasets created by these functions insert additional tape marks among the data component records thereby making it impossible to properly position a volume to any subsequent subdataset.

Prime Datasets

The first dataset on the first reel of a multi-dataset group is called the **prime dataset**. The prime dataset "owns" all the volumes in the group and controls their physical attributes.

All the datasets that follow the prime dataset in the group are called **subdatasets**. Subdatasets must be associated with the prime dataset. BIM-EPIC supports two methods for this:

- Automatically (using the DSN Catalog)
- Explicitly (using JCL)

If the datasets are associated using TLBL statements, the length of the prime dataset name is limited to 17 characters, although the subdataset names are not limited. This type of association is made by specifying the prime dataset name when the subdataset names are controlled.

If the datasets are associated using DD statements, the length of the prime dataset name can be up to 44 characters. This type of association is made by a referback to the prime dataset name on the DD statement for the subdataset. Therefore, a DD statement for the prime dataset name is necessary when the subdataset is created. This method allows the additional flexibility of associating the same subdatasets with different prime datasets depending on job requirements.

Subdatasets

Each subdataset has its own EDD, but some subdataset attributes cannot be specified because they are always inherited from the prime dataset. These attributes are:

- Density
 - Tape labeling
 - Tape size
 - Tape pool
 - Owner ID
 - Vaulting method
-

Scratching Multi-Dataset Tapes

Multi-dataset tapes are scratched when all active datasets on the tape are scratched. None of the volumes in a multi-dataset/multi-volume group is eligible for scratch until all active datasets on all active volumes are scratched.

DSN Catalog

1. If you want the group to be written to a particular tape pool, or with a particular density, or you want to specify any other physical property or handling of the tape, these characteristics must be specified in the prime definition. Applicable parameters on the TSIDMNT CATALOG statement include:

- density (MOD)
- label type (LBL)
- owner ID (OWN)
- tape size (SIZ)
- vaulting method (VLT)
- tape pool (TPL)

2. Several important limitations apply when creating a subdataset EDD.

You must catalog the subdatasets using Online Manager (or TSIDMNT), or autocatalog them using DD statements. You cannot use TLBL statements to autocatalog subdatasets.

You must enter the prime dataset name correctly the first time for a subdataset EDD, because you cannot change the prime dataset name field in an existing subdataset EDD.

You can supply independent retention criteria, passwords, and block sizes for each subdataset.

You may find it helpful to give all the datasets in each multi-dataset group a single, unique user ID or system ID. This makes it easier to report or scratch them as a group, rather than as individual datasets.

JCL Requirements

1. Inhibit LUB allocation for all datasets using either of the following:

- option "64" in a TLBL statement
- SYSnnn option in a DD statement

If you are creating subdatasets with a single program that opens multiple DTFs with different logical units, you must inhibit LUB allocation through the DSN Catalog. To force BIM-EPIC to use the same logical unit for each dataset, specify the same SYS number in each EDD.

If you are creating subdatasets with different programs using DTFs with different logical units, you can use a VSE // ASSGN statement to assign the SYS number from the previous step to the SYS number in the current step. This may be necessary when using different utility programs to create subdatasets with different SYS numbers.

2. BIM-EPIC corrects the file sequence number on your TLBL, if it is incorrect. You may omit the file sequence number.

3. If your program uses multiple open requests for the same DTF name, you must specify TLBL option 16 (dataset drop). Dataset drop is automatic when you use multiple DD statements with the same ddname.

4. Prevent rewind on close for each dataset processed except the last. This can be accomplished with any of the following:

- option code "3" in a TLBL statement
- VOL=(,RETAIN) in a DD statement
- OPT=NOREW in a DD statement

For the last dataset, you may want to specify a rewind or unload option code. Rewind can be accomplished with either of the following:

- option "1" (rewind) in a TLBL statement
- FREE=CLOSE (rewind) in a DD statement
- Unload can be accomplished with either of the following:
- option "2" (unload) in a TLBL statement
- VOL=(PRIVATE) (unload) in a DD statement

5. For each dataset processed except the last, prevent the release of assignment. This can be accomplished with either of the following:

- option "8" in a TLBL statement
- FREE=END in a DD statement

For the last dataset, you usually want the assignment to be released at close. This can be accomplished by omitting the option code.

6. On output, BIM-EPIC continues to write new datasets at the end of the current multi-dataset group. When you want to create a new group, you must start by creating a new version of the prime dataset.

7. On input, you can process the datasets in any order. Because of the physical limitations of tape media, it is most efficient to process the datasets in ascending order by file sequence number, but this is not mandatory.

8. To automatically catalog a subdataset, you must use DD statements.

If the length of the prime dataset name is 17 characters or less, the prime dataset name is entered in the Comment field of the subdataset EDD. If the prime dataset name in the comment field is longer than 17 characters, BIM-EPIC puts XXXXXXXXXXXXXXXXXXXX in the subdataset's Comment field.

Be sure the dataset is not already controlled before you attempt to catalog it using this option. If the prime dataset or any subdataset is already controlled, DISP=(x,CATLG) is ignored.

Multi-Dataset Processing Examples

Example 1

The first example is one of the simplest possible situations for creating a multi-dataset group. Your program creates datasets with different DTF names, but each DTF uses the same logical unit. The datasets are controlled in the first step. They are created in the second step. The second dataset on the tape is read in the third step. In the cataloging step, the prime dataset is defined first. It is given a tape pool which applies to all its subdatasets. All other datasets are defined with the NSU parameter, which makes them subdatasets. SUB1 and SUB2 have their own retention parameters specified. SUB3 is assigned the BIM-EPIC system's default retention. All datasets are defined with the user ID TS, so that they can be easily identified as a group. The last dataset does not require the tape and drive to be held, nor does it require dataset drop.

```
// JOB XYZ
*
* -----+
*  CATALOG A PRIME DSN AND THREE SUB DSN'S      +
* -----+
*
// EXEC TSIDMNT
CAT 'PRIME' CYC=2 UID=TS TPL=A
CAT 'SUB1' NSU='PRIME' RET=14 UID=TS
CAT 'SUB2' NSU='PRIME' CYC=3 UID=TS
CAT 'SUB3' NSU='PRIME' UID=TS
/*
*
* -----+
*  CREATE A PRIME AND 3 SUB DSN'S.  PROGX PERFORMS 4 OPENS, +
*  ONE EACH FOR TAPE1, TAPE2, TAPE3 AND TAPE4.             +
* -----+
*
// TLBL TAPE1,'PRIME',,,,,,75          64 + 8 + 3
// TLBL TAPE2,'SUB1',,,,,,75           64 + 8 + 3
// TLBL TAPE3,'SUB2',,,,,,75           64 + 8 + 3
// TLBL TAPE4,'SUB3',,,,,,65           64 + 1
// EXEC PROGX
/*
*
* -----+
*  READ FILE=2 ONLY                                         +
* -----+
*
// TLBL TAPEIN,'SUB1'
// EXEC PROGY
/*
/&
```

Options Used

Code	Action
1	Rewinds the tape (for the last dataset)
3	Inhibits rewind at close (for all but the last dataset)
8	Holds the drive assignment at close (for all but the last dataset)
64	Inhibits LUB allocation (for all datasets)

Example 2

This example creates multiple datasets with the same DTF name. This process requires dataset drop.

```
// JOB XYZ
*
* -----+
* CATALOG A PRIME AND 2 SUB DSN'S                +
* -----+
*
// EXEC TSIDMNT
CAT 'PRIME' CYC=3 UID=KM
CAT 'SUB1' NSU='PRIME' UID=KM CYC=4
CAT 'SUB2' NSU='PRIME' UID=KM RET=14
/*
*
* -----+
* CREATE A PRIME AND 2 SUB DSN'S.  PROGX PERFORMS+
* 3 OPENS FOR THE DTFNAME TAPEOUT.                +
* -----+
*
// TLBL TAPEOUT,'PRIME',,,,,,91          64 + 16 + 8 + 3
// TLBL TAPEOUT,'SUB1',,,,,,91          64 + 16 + 8 + 3
// TLBL TAPEOUT,'SUB2',,,,,,66          64 + 2
// EXEC PROGX
/*
*
* -----+
* READ A PRIME AND 2 SUB DSN'S.  PROGY PERFORMS  +
* 3 OPENS FOR THE DTFNAME TAPEOUT.                +
* -----+
*
// TLBL TAPEIN,'PRIME',,,,,,91          64 + 16 + 8 + 3
// TLBL TAPEIN,'SUB1',,,,,,91          64 + 16 + 8 + 3
// TLBL TAPEIN,'SUB2',,,,,,66          64 + 2
// EXEC PROGY
/*
*
*&
```

The options in use are:

Code	Action
1	Rewinds the tape (for the last dataset)
2	Dismounts the tape (for the last dataset)
3	Inhibits rewind at close (for all but the last dataset)
8	Holds the drive assignment at close (for all but the last dataset)
16	Drops the TLBL (for all but the last dataset)
64	Inhibits LUB allocation (for all datasets)

The last dataset does not require the tape and drive to be held, nor does it require dataset drop.

Example 3

This example creates multiple datasets with different DTF names. Because the different DTFs use different logical units, a logical unit must be supplied in the dataset definitions.

```
// JOB XYZ
*
* -----+
* CATALOG A PRIME AND 2 SUB DSN'S +
* -----+
*
// EXEC TSIDMNT
CAT 'PRIME' SYS010
CAT 'SUB1' NSU='PRIME' SYS010
CAT 'SUB2' NSU='PRIME' SYS010
/*
*
* -----+
* CREATE A PRIME AND 2 SUB DSN'S.  PROGX PERFORMS 3 OPENS, +
* ONE EACH FOR TAPE1, TAPE2 AND TAPE3. +
* -----+
*
// TLBL TAPE1,'PRIME',,,,,,11          8 + 3
// TLBL TAPE2,'SUB1',,,,,,11          8 + 3
// TLBL TAPE3,'SUB2',,,,,,2          2
// EXEC PROGX
/*
*
/ &
```

The options used are:

Code	Action
2	Dismounts the tape (for the last dataset)
3	Inhibits rewind at close (for all but the last dataset)
8	Holds the drive assignment at close (for all but the last dataset)

Because LUB allocation is inhibited through the dataset definition (SYS010 specified in the definitions of both the prime dataset and the subdatasets), 64 does not have to be specified on the TLBL.

Example 4

This example is very similar to Example 2. It may seem more complicated than Example 2, but the BIM-EPIC DD facility offers greater flexibility than TLBLs.

```
// JOB XYZ
* -----+
*   CATALOG A PRIME DSN AND THREE SUB DSN'S      +
* -----+
*
// EXEC TSIDMNT
CAT 'PRIME'          CYC=2
CAT 'SUB1' NSU='PRIME' CYC=2
CAT 'SUB2' NSU='PRIME' CYC=2
/*
*
* -----+
*   CREATE A PRIME AND 2 SUB DSN'S USING BIM-EPIC DD FACILITY. +
*   PROGX PERFORMS 3 OPENS FOR DTFNAME TAPEOUT.              +
* -----+
*
* //TAPEOUT DD DSN='PRIME(+0)',
* //          OPTION=(SYS010),
* //          UNIT=TAPE,
* //          FREE=END,
* //          DISP=(NEW,KEEP),
* //          VOL=(,RETAIN)
* //TAPEOUT DD DSN='SUB1(+0)',
* //          OPTION=(SYS010),
* //          UNIT=TAPE,
* //          FREE=END,
* //          DISP=(NEW,KEEP),
* //          VOL=(,RETAIN)
* //TAPEOUT DD DSN='SUB2(+0)',
* //          OPTION=(SYS010),
* //          UNIT=TAPE,
* //          DISP=(NEW,KEEP),
* //          VOL=(PRIVATE)
// EXEC PROGX
/*
* -----+
*   READ A PRIME AND 2 SUB DSN'S USING BIM-EPIC DD FACILITY. +
*   PROGY PERFORMS 3 OPENS FOR DTFNAME TAPEIN.              +
* -----+
* //TAPEIN DD DSN='PRIME(0)',
* //          OPTION=(SYS010),
* //          UNIT=TAPE,
* //          FREE=END,
* //          DISP=(OLD,PASS),
* //          VOL=(,RETAIN)
* //TAPEIN DD DSN='SUB1(0)',
* //          OPTION=(SYS010),
* //          UNIT=TAPE,
* //          FREE=END,
* //          DISP=(OLD,PASS),
* //          VOL=(,RETAIN)
* //TAPEIN DD DSN='SUB2(0)',
* //          OPTION=(SYS010),
* //          UNIT=TAPE,
* //          DISP=(OLD,KEEP)
// EXEC PROGY
/*
/&
```

In the example above, the options used in the creation step are:

Option	Action
DSN='xxxx(+0)'	The "+0" in parentheses indicates that this is a generation dataset (see the DDGEN configuration option in the <i>BIM-EPIC Installation and System Guide</i>).
OPTION=(SYS010)	Inhibits LUB allocation for this dataset. This is equivalent to TLBL option code 64. In addition, this forces the use of SYS010 (which has no TLBL equivalent).
UNIT=TAPE	Tells BIM-EPIC to create a tape label. This would be determined in standard VSE JCL by using a TLBL.
FREE=END	Holds the assignment at close for all but the last dataset processed. This is equivalent to TLBL option code 8.
DISP=(NEW,KEEP)	The NEW subparameter indicates that this dataset version will be created as output. The KEEP subparameter causes the tape to be unloaded at close unless VOL=(,RETAIN).
VOL=(,RETAIN)	Inhibits rewinding the tape at close (overrides the DISP subparameter KEEP). This is equivalent to TLBL option code 3.
VOL=(PRIVATE)	Causes the tape to be unloaded at close. This is equivalent to TLBL option code 2.

The option used in the read step are:

Option	Action
DSN=' xxxx(0)'	The zero (0) in parentheses indicates that the current version (version 1) is to be processed.
DISP=(OLD,PASS)	The OLD subparameter indicates that this dataset version already exists and will be read as input. The PASS subparameter indicates that this tape is used later in the step or in a subsequent step.

Example 5

Whenever a subdataset such as `VERY.LONG.NAME.FOR.NEWSUB.3` is accessed, the prime dataset name must be referenced in the JCL. The following example shows how `VERY.LONG.NAME.FOR.NEWSUB.3` can be read as input:

```
// JOB XYZ
*
* -----+
* READ VERY.LONG.NAME.FOR.NEWSUB.3 AS INPUT.  THE PRIME DATASET +
* NAME (VERY.LONG.NAME.FOR.PRIME.1) MUST BE SPECIFIED AND  +
* REFERRED TO BY THE DD FOR THE SUBDATSET.  PROGY+
* PERFORMS 1 OPEN FOR DTFNAME TAPEIN.
* -----+
*
* //DUMMY    DD   DSN='VERY.LONG.NAME.FOR.PRIME.1(0)',
* //          UNIT=TAPE
* //TAPEIN   DD   DSN='VERY.LONG.NAME.FOR.NEWSUB.3(0)',
* //          UNIT=TAPE,
* //          DISP=(OLD),
* //          VOL=REF=*.DUMMY
// EXEC PROGY
/*
/ &
```

Note the use of the DTF name `DUMMY` in DD statements. This is also known as the ddname. Any ddname except `TAPEIN` can be used because `PROGY` performs an open for `TAPEIN`, not `DUMMY`.

Types of Datasets

Sequential Datasets

For disk, allocation is on track boundaries for CKD devices. If you specify DLBL option "A" or the SPACE=(...,ROUND) parameter on a DD statement, BIM-EPIC makes the allocation on cylinder boundaries.

Otherwise, there are no special requirements for sequential dataset processing.

Direct Access Datasets

Description

For direct access datasets:

- LUB allocation is inhibited automatically
- Allocation is on cylinder boundaries and datasets are not truncated
- Secondary extents are not supported for controlled datasets

Configuration Options

For *uncontrolled* DA datasets, the following apply:

- When UNCDISK=NO, there is no limit to the number of extents.
- When UNCDISK=YES, there is a limit of 5 extents.

DSN Catalog

No special EDD requirements.

JCL Requirements

The VSE job control program requires you to use a DLBL and a complete EXTENT statement. For BIM-EPIC purposes, however, only the file type (access method "DA") on the DLBL and the extent sequence numbers on the EXTENT statement must be correct. For both cataloged and Start Track 1 datasets, the extent sequence numbers must begin with 1. VSE also requires that secondary extents reside on separate volumes.

Examples

The following example shows the EXTENT statement for a controlled dataset. BIM-EPIC supplies the volume serial number, starting address, and number of tracks/blocks from information in the catalog.

```
// DLBL RESDA,'dataset.name',,DA
// EXTENT SYS020,,1,1
```

The following example shows the EXTENT statement for a Start Track 1 dataset. BIM-EPIC supplies the actual volume serial number and starting address.

```
// DLBL RESDA,'dataset.name',,DA
// EXTENT SYS020,POOL01,1,1,1,15
```

On BIM-EPIC DD statements, specify DCB=DSORG=DA or DCB=DSORG=DAU. This example could be used to read or update a controlled DA dataset.

```
* //RESDA DD DSN=INVOICE.TICKETS,DISP=(OLD,KEEP),
* // DCB=DSORG=DA
```

ISAM Datasets

Description

BIM-EPIC can control ISAM datasets. The following VSE restrictions affect ISAM dataset processing with BIM-EPIC:

- BIM-EPIC can control only one prime data area, one overflow area, and one index per ISAM dataset.
- ISAM datasets are automatically allocated on cylinder boundaries.
- If UNCDISK=YES is specified as a configuration option, or if you specify DLBL options in your JCL, uncontrolled ISAM datasets are limited to a maximum of 5 extents total.

DSN Catalog

There are no special requirements for the catalog entry. Extent size specified in the catalog refers to the prime data area. Multiple extents are not allowed. Extent information for the index and overflow areas is taken from JCL. The disk pool applies to all areas, prime data, index, and overflow.

JCL Requirements

EXTENT statements are always required for all three areas, as follows:

1. For catalog controlled datasets, the number of tracks to allocate for the prime area is taken from the catalog, unless you specify the DLBL 'O' option. The number of tracks to allocate for index and overflow areas must be specified on the appropriate extent statement, as shown in the following two examples:

```
// DLBL MASTER, 'EPIC.ISAM.MASTER.FILE', ,ISE
// EXTENT SYS040,,4,1      Index
// EXTENT SYS041,,1,2      Prime
// EXTENT SYS042,,2,4      Overflow
```

```
// DLBL MASTER, 'EPIC.ISAM.MASTER.FILE', ,ISC
// EXTENT SYS040,,4,1,,15  Index
// EXTENT SYS041,,1,2      Prime
// EXTENT SYS042,,2,4,,30  Overflow
```

2. For Start Track 1, the SYS number for the overflow extent must be different from the SYS number for the index and prime data areas, as shown in the following example:

```
// DLBL MASTER, 'EPIC.ISAM.MASTER.FILE', ,ISE
// EXTENT SYS040,POOL80,4,1,1,15  Index
// EXTENT SYS041,POOL80,1,2,1,90  Prime
// EXTENT SYS042,POOL80,2,3,1,30  Overflow
```

3. For each DD statements, specify DCB=DSORG=IS or DCB=DSORG=ISU.

The three areas of an ISAM dataset (prime, index, and overflow areas) are all defined as part of the same DD statement. However, the overflow area must be defined using its own parameters (DSN, DCB, DISP, SPACE, VOL, and so on). You can define the prime and index areas together (with one set of parameters) or independently.

Examples

In the following example, each area is defined with its own set of parameters. Space and other information about the area follows the DSN parameter identifying the area. Because it is defined alone, index area space is defined with the primary-qty subparameter, SPACE=(TRK,1) and not the index-qty, SPACE=(TRK(,1)).

```
* //DATA      DD  DSN=CUSTOMER.MASTER( PRIME ) ,
* //          DISP=(NEW,KEEP)
* //          UNIT=DISK
* //          DCB=DSORG=IS ,
* //          SPACE=( TRK , 45 )
* //          VOL=SER=POOL01
* //          DD  DSN=CUSTOMER.MASTER( INDEX ) ,
* //          DISP=(NEW,KEEP)
* //          UNIT=DISK ,
* //          DCB=DSORG=IS ,
* //          SPACE=( TRK , 1 )
* //          VOL=SER=POOL01
* //          DD  DSN=CUSTOMER.MASTER( OVFLOW ) ,
* //          DISP=(NEW,KEEP)
* //          UNIT=DISK ,
* //          DCB=DSORG=IS ,
* //          SPACE=( TRK , 15 )
* //          VOL=SER=POOL01
```

In the following example, one set of parameters is used to define both the prime and the index areas. In this case, SPACE is defined for the index area by using the index-qty subparameter.

```
* //DATA  DD  DSN=CUSTOMER.MASTER ,
* //          DISP=(NEW,KEEP) ,
* //          UNIT=DISK ,
* //          DCB=DSORG=IS ,
* //          SPACE=( TRK , ( 45 , , 1 ) ) ,
* //          VOL=SER=POOL01
```

VSAM and VSAM-Managed SAM Datasets

Description

BIM-EPIC does not control VSAM or VSAM-managed SAM datasets. Therefore no changes are required in your JCL. However, you can use DD statements to access an existing VSAM cluster or VSAM-managed SAM dataset. See the IBM documentation for further information.

You can have VSAM dataset location and activity recorded in the DSN Catalog. If you catalog the VSAM cluster using Online Manager (or TSIDMNT), this information will appear on the BIM-EPIC reports and displays, just as it does for non-VSAM datasets. At close time, BIM-EPIC logs the allocations used for any suballocated or unique VSAM clusters.

Configuration Options

The VSAM configuration option enables logging of VSAM datasets.

Work Datasets

Temporary Datasets

BIM-EPIC can release disk space or tapes used for controlled datasets at close or at end-of-job.

You can find common work dataset (CWDS) support or CPU/partition independence helpful with work datasets. See "Common Work Datasets" and "CPU and Partition Independence" for details.

DSN Catalog

Catalog the EDD as work ("WRK" using Online Manager, or WRK=YES using TSIDMNT).

JCL Requirements

For TLBLs, specify option "4".

For DLBLs, specify option "D", "P", or "X" in the dataset name.

For DD statements, specify DISP=(NEW,DELETE,DELETE).

CPU and Partition Independence

Description CPU and partition independence are normally used with work datasets. They allow you to use a single set of JCL to run in any partition or CPU. Because BIM-EPIC inserts a partition and/or CPU ID into the dataset name to make the dataset name unique, there will be no duplication of the dataset names in different partitions or machines.

For example, in your JCL you could specify `===.SORTWRK1` as the dataset name. The three equal signs designate both CPU and partition independence. When you run the job in the BG partition of CPU ID 1, BIM-EPIC uses the dataset name 'BG1.SORTWRK1'. When you run it in the F1 partition of CPU ID 2, BIM-EPIC uses the name 'F12.SORTWRK1'.

DSN Catalog Catalog a separate dataset definition for each partition/CPU ID combination you intend to use. Specify the partition/CPU ID first, then a period, and then the dataset name. For the example above, you would create a definition for BG1.SORTWRK1 and F12.SORTWRK1. When using both CPU and partition independence, the partition ID precedes the CPU ID.

Note: If you actually specify `===` in the dataset name when it is controlled, the `===` is replaced by the partition and CPU ID in which the catalog operation occurs.

JCL Requirements 1. For both DLBL and DD statements, prefix your dataset name with equal signs (=) and a period (.), as follows:

Format	Specifies...
<code>=.dataset.name</code>	CPU independence only
<code>==.dataset.name</code>	Partition independence only
<code>===.dataset.name</code>	Both CPU and partition independence

2. For DD statements, you can substitute two ampersands (&&) for the equal signs and omit the period. If you are concerned about MVS compatibility, using the ampersands can be an advantage because they signify a temporary dataset to MVS. Note that MVS requires that ampersands be specified as the first two characters of the dataset name. See the DSN parameter for more information.

Example This dataset will have a unique name in any partition or CPU in which it is used.

```
// DLBL WORKOUT, '===.WORKFILE.1'
```

Common Work Datasets (CWDS)

Description

The easiest way to control sort work, compiler work areas and any other disk only dataset that does not need to be kept after end-of-job is to use the common work dataset (CWDS) feature.

When a dataset name begins with three dollar signs and a period (\$\$\$., or £££. in the U.K.), BIM-EPIC recognizes it as a CWDS. The \$\$\$ (or £££) in the dataset name are replaced by the CPU ID and partition ID in which the job is executed. This substitution is the reverse of "===" datasets, to avoid any conflicts if both are used with the same dataset name. In addition, a CWDS is automatically deleted at end-of-job, even if the job cancels.

Since each version is deleted at the end of every job, no versions are recorded in the DSN Catalog or kept on disk. If you attempt to open a version as input in a new job, the job cancels with an appropriate error message.

CWDS requires that you catalog only one dataset name, unlike CPU/partition independence which requires you to catalog a partition/CPU independent dataset name for every partition and machine combination in which the dataset could be used.

DSN Catalog

1. Catalog the dataset name beginning with '\$\$\$.' ('£££.' in the U.K.). Use the remaining 40 characters to name the dataset.
2. Specify as work ("WRK" using Online Manager, or WRK=YES using TSIDMNT).
3. Cycle, retention period, and DLA retention are ignored if you specify them.

Example

This control statement catalogs a CWDS using TSIDMNT.

```
CAT   '$$$ . SORTWK1 ' , WRK=YES , POL=POOL01 , EXT=120
```

JCL Requirements

Specify the dataset name beginning with '\$\$\$' ('£££' in the U.K.). No delete options are necessary on the dataset's DLBL or DD statement.

TYPEFLE=WORK DTFs

Description

Datasets that have TYPEFLE=WORK DTFs are processed differently from other sequential datasets. Common TYPEFLE=WORK DTFs include compiler work datasets and FORTRAN disk datasets.

BIM-EPIC can process TYPEFLE=WORK DTFs on disk only (DTFSD). You can, of course, still use TYPEFLE=WORK DTFs with tape if the tape dataset is uncontrolled.

TYPEFLE=WORK DTFs can be controlled as Start Track 1 or cataloged datasets. However, we strongly recommend that you use the Catalog to eliminate many allocation problems. For example, BIM-EPIC automatically reuses the same extents used by an earlier version of a cataloged dataset, if one exists. If you use Start Track 1 datasets, the old dataset is deleted the next time a dataset with the same dataset name is opened.

Secondary extents are not allowed.

DSN Catalog

1. Catalog the dataset as work ("WRK" using Online Manager, or WRK=YES using TSIDMNT).
 2. You can inhibit LUB allocation for the dataset by specifying a SYS number in the Catalog.
-

JCL Requirements

Inhibit LUB allocation if this is not done through the Catalog. For DLBLs, specify the DLBL option "N" in the dataset name. For DD statements, include the OPTION=SYSnnn parameter.

Compiler Work Datasets

Description

Compiler work datasets have the same requirements as TYPEFLE=WORK DTF datasets. Refer to "TYPEFLE=WORK DTF's" above, for those requirements.

In addition, observe the following requirements described below.

DSN Catalog

Compiler work datasets should be controlled with a disk pool that contains volumes of the same device type as those to which SYS001, SYS002, SYS003, SYS004, and SYSLNK are permanently assigned.

JCL Requirements

1. Each compiler does a pre-open check to determine the device type prior to opening its work datasets. Logical units SYS001, SYS002, SYS003, SYS004, and SYSLNK must therefore be pre-assigned to the same device type to be used for the compiler work datasets. You can use permanent or temporary assignments to accomplish this.
2. All compiler work dataset labels must be submitted before the compiler is executed.
3. LUB allocation is inhibited automatically, since the DTF names begin with IJSYS.

Example

```
// JOB COMPILE AND LINK EDIT
// ASSGN SYS001,SYSWK1
// ASSGN SYS002,SYSWK1
// ASSGN SYS003,SYSWK1
// ASSGN SYS004,SYSWK1
// ASSGN SYSLNK,SYSWK1
// DLBL IJSYS01,'$$$ .IJSYS01'
// EXTENT SYS001
// DLBL IJSYS02,'$$$ .IJSYS02'
// EXTENT SYS002
// DLBL IJSYS03,'$$$ .IJSYS03'
// EXTENT SYS003
// DLBL IJSYS04,'$$$ .IJSYS04'
// EXTENT SYS004
// DLBL IJSYSLN,'$$$ .IJSYSLN'
// OPTION CATAL
// PHASE PROGX,*
// EXEC FCOBOL
// (COBOL SOURCE)
/*
// EXEC LNKEDT
/&
```

Uncontrolled Datasets

Description

Any dataset that is not controlled using the DSN Catalog or the Start Track 1 features is called an uncontrolled dataset. With a few exceptions, uncontrolled datasets are not provided any additional features than are provided using standard VSE.

For uncontrolled tape datasets, BIM-EPIC provides protection for all output tapes to ensure that any tape volume that is still in an active status is not used. Additionally, if the configuration option UNCTL is set to NO, uncontrolled tape datasets cannot use any tape in the DSN Catalog even if it is in a scratch status. VSE ASSGN statements are not required for uncontrolled tape datasets. If there is no assignment, the operator is prompted to specify an available tape drive which can be used. Unlike controlled tape datasets, the assignment is not released at close unless the configuration option UNCTAP is set to YES.

Uncontrolled disk datasets can be placed within the extents of a disk pool provided they are created with adequate retention. ASSGN and EXTENT statements are normally required, but if the configuration option UNCDSK is set to YES, the ASSGN statement is optional.

JCL Requirements

No special options are required for uncontrolled datasets.

Physical IOCS Processing

Introduction

VSE system utilities, sort programs, report writers, database utilities, and many other programs process datasets with physical IOCS (PIOCS).

Many different issues can come into play when a program uses PIOCS to process datasets that are under BIM-EPIC control. If you familiarize yourself with those described below, you can make any PIOCS program work with BIM-EPIC-controlled datasets. When you are familiar with these points, please review the sections on program-specific instructions.

PIOCS Issues

1. When a program requires a specific logical unit, LUB allocation must be inhibited for every dataset processed in that job step. There are some cases in which LUB allocation is inhibited automatically. See "LUB Allocation and the ASSGN Statement" in this chapter for details.

2. Satisfy pre-open checking:

- a. Make an assignment to a device type, using any of the following techniques:

- make a permanent assignment, as for SYS001 through SYS005 and SYSLNK
- make a temporary assignment
- use the DD OPTION=SYSnnn parameter
- execute TSIDSRT (for SORT) or TSIDASS ASSGN

If you make a disk assignment using the VSE ASSGN statement, assign the dataset to a device of the same type to be used for the dataset. For example, you can assign a disk dataset with the following statement:

```
// ASSGN SYS003,DISK,VOL=SYSWK1,SHR
```

If SYSWK1 is a 3380, for example, your dataset can be created on any 3380 volume.

- b. For other programs like DITTO, some IDCAMS functions, and some SORT products, include an EXTENT card with the assigned logical unit number.
- c. Many programs that use PIOCS do not issue an "open" request that BIM-EPIC can recognize. In this case, you can satisfy BIM-EPIC's "open" processing requirements by executing the TSIDASS OPEN function to simulate the opening of both input and output datasets. This is required only when the program does not issue an open, or does I/O to the dataset before it is opened. Examples include some DITTO functions, LIBR stand-alone backups and most FORTRAN compilers.
3. Use caution when using BIM-EPIC's concatenation facility. BIM-EPIC cannot recognize end-of-file processing for many PIOCS programs, so BIM-EPIC's dataset concatenation may not work for these programs.
4. Most PIOCS programs do not honor BIM-EPIC's dataset reblocking facility. Alternate block sizes must usually be specified by submitting control statements to the program. For sorts, TSIDSRT can be used to automatically update the control statements with the BIM-EPIC block size.

5. Generic device independence (GDI) *cannot* be used with PIOCS.
 6. Some PIOCS programs do not provide the address of the last record written, so BIM-EPIC may not be able to truncate the dataset.
-

Utility Programs

BIM-EPIC provides the following utility programs to facilitate processing with PIOCS programs:

Program	Can be used for...
TSIDASS	Pre-open assignment checking and simulation of open processing
TSIDSRT	Sort programs
TSIDDEL	Deleting disk work datasets that have not been closed

These programs are discussed in detail in Chapter 8, “BIM-EPIC Utilities”

Sorting with BIM-EPIC

Introduction

Description

This section describes the considerations necessary to use BIM-EPIC controlled datasets with all sort programs in general and these three sort programs specifically:

- SYNCSORT using BIM-EPIC Interface
- IBM SORT beginning with DFSORT 3.3
- CA-SORT

All sort programs check the assignment of a particular logical unit (SYSnnn) number before actually opening a particular file (known as pre-open checking) and use PIOCS (Physical Input Output Control Systems) to open, close, read, and write their datasets. Therefore, some special considerations may be necessary when accessing BIM-EPIC controlled datasets with your sort program. These considerations fall into three categories:

- Satisfaction of pre-open checking
- Acquisition of necessary DTF information
- Deletion of the SORTWK dataset(s)

Pre-open checking can be satisfied using JCL considerations. BIM-EPIC automatically deletes controlled SORTWK datasets at EOJ. DTF information not supplied by the sort program either is lost or must be acquired by other means.

BIM-EPIC Sort Interface (TSIDSRT)

Description

The BIM-EPIC sort interface program (TSIDSRT) addresses all the considerations described above, and usually requires no JCL changes except the phase name on the EXEC statement. TSIDSRT does the following:

1. Reads the sort control statements and modifies the input BLKSIZE to match the block size in the BIM-EPIC Catalog.
2. Assigns tape datasets to a "dummy" tape device and opens disk datasets to satisfy pre-open checking.
3. Calls the sort program.
4. Uses the sort program's E35 exit routine to write the SORTOUT dataset. This allows access to DTF information necessary to support such BIM-EPIC options as disk dataset truncation, reblocking, and logging of block size, record size and record count.
5. Closes SORTWK datasets to enable them to be deleted.

Unless using a sort program that has a direct interface to BIM-EPIC (e.g. SYNCSORT), the recommended method of using sort programs with BIM-EPIC-controlled datasets is to use TSIDSRT. BIM-EPIC special option EPS083 provides for automatically changing 'SORT' to 'TSIDSRT' on EXEC statements.

Sort Specifics

Description

Many BIM-EPIC features rely upon DTF information that is automatically provided by the LIOCS (Logical Input Output Control Systems) access methods. BIM-EPIC also relies upon the standard open/close/end-of-extent/end-of-reel processing used by LIOCS. Not all sort programs use LIOCS to process data so the functionality of BIM-EPIC features varies depending upon which sort product is used

The following table details the operation of specific BIM-EPIC features when used with several sort programs. These considerations apply when the sort program is directly executed by using the VSE EXEC statement and then only to datasets that are opened directly the sort program (i.e., not those that may be opened by user-written sort exits). When sort is invoked internally by another program or TSIDSRT in directly executed, these considerations usually apply only to the SORTWK file.

BIM-EPIC feature	IBM SORT prior to DFSORT 3.3	IBM SORT DFSORT 3.3 and above	SYNCSORT without interface to BIM-EPIC	SYNCSORT with interface to BIM-EPIC	CA-SORT	TSIDSRT
Reblocking	No	No	No	Yes	No	Yes
Truncation of disk output dataset	No	No	No	Yes	No	Yes
Deletion of SORTWK (using close)	No	Yes	No	Yes	Yes	Yes
DTF information posting	No	No	No	Yes	No	Yes
LUB allocation	No	Yes	No	Yes	Yes	Based on sort program
Disk dataset concatenation	No	Yes	No	Yes	Yes	Based on sort program
Tape dataset concatenation	No	Yes	No	Yes	No	Based on sort program
Secondary SORTWK allocation	No	Yes	No	Yes	Yes	Based on sort program
Assignments required before SORTIN, SORTOUT or SORTWK opened?	Yes	No	Yes	No	No	No

**SYNCSORT using
BIM-EPIC Interface**

SYNCSORT offers options to improve compatibility with BIM-EPIC. See the *SYNCSORT Installation Guide* for details about these options. In addition, there are BIM-EPIC special options which must be activated when using SYNCSORT. See the *BIM-EPIC Installation and System Guide*, "Compatibility With Other Software" for details.

With these options activated, LUB allocation and secondary SORTWK allocations are supported, SORTWK datasets are closed to allow them to be deleted and disk output datasets are closed in a manner allowing for file truncation. In addition, SYNCSORT can communicate directly with BIM-EPIC to determine input file block sizes and to allow output file record and block sizes to be stored in the DSN Catalog and on tape HDR2 labels.

As a result, all of the problems typically associated to sort programs when using BIM-EPIC controlled datasets are resolved.

**IBM SORT
beginning with
DFSORT 3.3**

DFSORT offers an option to improve compatibility with BIM-EPIC. See the *DFSORT/VSE Installation and Tuning Guide*, Customizing DFSORT/VSE to interact with the File Management System installed at your site for details. With this option activated, LUB allocation and secondary SORTWK allocations are supported and input datasets can be concatenated. In addition, the method by which pre-open checking is accomplished is changed to checking the type of label that was submitted for a dataset to determine if it resides on tape or disk.

As a result, some of the problems typically associated to sort programs when using BIM-EPIC controlled datasets are resolved.

CA-SORT

CA-SORT offer an option to improve compatibility with BIM-EPIC. See the *CA-SORT Systems Programmer Guide* for details. With this option activated along with normal processing characteristics of this sort program, LUB allocation and secondary SORTWK allocations are supported and input disk (not tape) datasets can be concatenated. In addition, the method by which pre-open checking is accomplished is to check the type of label that is submitted for a dataset to determine if it resides on tape or disk.

As a result, some of the problems typically associated to sort programs when using BIM-EPIC controlled datasets are resolved.

SORTWK Datasets

Description	To provide optimal utilization of disk work space, SORTWK datasets should be defined as temporary datasets and deleted as soon as they are no longer in use.
DSN Catalog	<ol style="list-style-type: none"> 1. Catalog SORTWK datasets as work ("WRK" using Online Manager, or WRK=YES using TSIDMNT). 2. Use the CWDS facility or partition/CPU-independent datasets so that the datasets can be processed in any partition without conflict.
JCL Requirements	<p>Various BIM-EPIC options can be used to delete SORTWK areas upon completion of a sort.</p> <ol style="list-style-type: none"> 1. Controlled SORTWK datasets are deleted automatically at end of job. 2. Common work datasets (\$\$\$) are also deleted at end of job. 3. If the sort program closes SORTWK, use the DLBL "D" option or the DD DISP=(,DELETE) parameter to delete SORTWK at close. 4. If the sort program does not close SORTWK, use the DLBL "D" option or the DD DISP=(,DELETE) parameter and execute TSIDDEL after every sort step. <p>For jobs that contain multiple sort steps, SORTWK datasets should be deleted after each sort step, using either (3) or (4) above as applicable.</p>
Pre-Open Checking	<p>All sorts check specific SYS numbers to determine the device type for each dataset that is to be opened. The sort searches the following sources (in the order listed) for the SYS number to be used for each dataset. The search stops as soon as a SYS number is found.</p> <ol style="list-style-type: none"> 1. The SYS number found in the VSE label area for disk datasets (from the EXTENT statement or the DD OPTION=SYSnnn parameter) 2. Sort control statements (SORTIN, SORTOUT, SORTWK keywords); which TSIDSRT can modify if necessary. 3. Sort defaults (standard or user-specified: SYS001 for SORTOUT, SYS002 for SORTIN1, SYS003 for SORTWK, and so on). <p>NOTE: If LUB allocation is active for sort datasets, the SYS numbers checked by the sort in its pre-open processing may not be the same SYS numbers assigned by BIM-EPIC when the open actually occurs.</p>

JCL Requirements

Pre-open assignments can be met by any of the following means:

Media	Method
Disk	1. Permanent assignments (in partition start-up procedure) 2. Temporary assignments (in each job)
Tape	Execute TSIDASS with the ASSGN function.
Tape or Disk	1. Use the DD OPTION=SYSnnn parameter. 2. Execute TSIDSRT.

One popular method of satisfying pre-open requirements for the sort (and compilers) is to permanently assign SYS001 through SYS005 to disk in every partition in which the compiler may be used. The disk assigned must be the same device type as the device type in the BIM-EPIC disk pool.

Examples

Many sample job streams for sort applications are supplied when BIM-EPIC is installed. Please see the *BIM-EPIC Installation and System Guide*, "BIM-EPIC Sample Jobstreams".

IDCAMS BACKUP and RESTORE

Description	You can use BIM-EPIC to control the IDCAMS backup dataset produced by the BACKUP function and used by the RESTORE function. IDCAMS provides for the backup dataset to reside on either tape or disk.
DSN Catalog	Only prime dataset definitions are allowed. Therefore, multi-dataset tape groups are specifically disallowed. For disk only, non-generation single-extent disk datasets (WRK=YES) residing on CKD devices only may be used and truncation must be bypassed.
JCL Requirements	<p>Each controlled tape dataset must be defined with a TLBL or DD statement and each controlled disk dataset must be defined with a DLBL and abbreviated EXTENT statement (// EXTENT SYS00n) or a DD statement with the OPTION=SYS00n parameter.</p> <p>LUB allocation must be inhibited since IDCAMS requires SYS005 be used for the output backup dataset and SYS004 for the input backup dataset.</p>
Restrictions	Due to IDCAMS restrictions, you cannot use GDI to route a tape backup to disk or vise-versa. When using disk for the backup dataset, a new extent must be allocated and the correct assignment made before starting the backup function.
Examples	Many sample job streams for IDCAMS BACKUP and RESTORE are supplied when BIM-EPIC is installed. Please see the <i>BIM-EPIC Installation and System Guide</i> , "BIM-EPIC Sample Jobstreams".

IDCAMS REPRO

Description	You can use BIM-EPIC to control non-VSAM sequential datasets for both input and output with the IDCAMS REPRO function.
DSN Catalog	No restrictions exist. For disk, both generation and non-generation datasets with single or multiple extents may be used. For tape, both single and multi-dataset groups residing and single or multiple tape volumes may be used.
JCL Requirements for Tape	<p>Each controlled dataset must be defined with a TLBL or DD statement.</p> <p>LUB allocation must be inhibited since IDCAMS requires SYS005 to be used for output datasets and SYS004 to be used for input datasets.</p>
JCL Requirements for Disk	Each controlled dataset must be defined with a DLBL and abbreviated EXTENT statement (// EXTENT SYS00n) or a DD statement with the OPTION=SYS00n parameter. Only when using a DLBL and EXTENT, the SYS number chosen must be assigned to any disk. Although you can use the SYS number of your choice, typically SYS001 is an easy choice since most installations have SYS001 permanently assigned to disk.
Examples	Many sample job streams for IDCAMS REPRO are supplied when BIM-EPIC is installed. Please see the <i>BIM-EPIC Installation and System Guide</i> , "BIM-EPIC Sample Jobstreams".

IDCAMS IMPORT/EXPORT

Description	You can use BIM-EPIC to control the portable file produced by the EXPORT function and read by the IMPORT function.
DSN Catalog	No restrictions exist. For disk, both generation and non-generation datasets with single or multiple extents may be used. For tape, both single and multi-dataset groups residing and single or multiple tape volumes may be used.
JCL Requirements for Tape	<p>Controlled portable files must be defined with a TLBL or DD statement.</p> <p>LUB allocation must be inhibited since IDCAMS requires SYS005 to be used when creating a portable file and SYS004 to be used when reading a portable file.</p>
JCL Requirements for Disk	<p>Each controlled dataset must be defined with a DLBL and abbreviated EXTENT statement (// EXTENT SYS00n) or a DD statement with the OPTION=SYS00n parameter. Only when using a DLBL and EXTENT, the SYS number chosen must be assigned to any disk .</p> <p>Although you can use the SYS number of your choice, typically SYS001 is an easy choice since most installations have SYS001 permanently assigned to disk.</p>
Examples	Many sample job streams for IDCAMS EXPORT and IMPORT are supplied when BIM-EPIC is installed. Please see the <i>BIM-EPIC Installation and System Guide</i> , "BIM-EPIC Sample Jobstreams".

LIBR BACKUP/RESTORE

Description	You can use BIM-EPIC to control the tape datasets created by the BACKUP function and read by the RESTORE function.
DSN Catalog	Only prime dataset definitions are allowed. Therefore, multi-dataset tape groups are specifically disallowed.
JCL Requirements	<p>Each controlled dataset must be defined with a TLBL or DD statement.</p> <p>LUB allocation must be inhibited since LIBR requires use of the SYS number specified in the BACKUP or RESTORE control statement. In addition, the tape drive assignment must be held after the dataset is closed since LIBR may attempt to access the tape drive even after the close has been accomplished.</p> <p>Only when including the Standalone Restore programs (RESTORE=STANDALONE) with the BACKUP function, the first (or only) tape output volume must be allocated before the BACKUP function is attempted by using the TSIDASS OPEN function.</p>
Control Statements	Use the TAPELABEL parameter to specify a one to seven alphanumeric filename, the first character of which must be alphabetic. This is the same filename reflected in the accompanying TLBL or DD statement.
Restrictions	Due to LIBR restrictions, you cannot use GDI to route a tape backup to disk.
Examples	Many sample job streams for LIBR BACKUP and RESTORE are supplied when BIM-EPIC is installed. Please see the <i>BIM-EPIC Installation and System Guide</i> , "BIM-EPIC Sample Jobstreams".

DITTO

Description	<p>You can use BIM-EPIC to control tape and sequential disk files logically accessed by DITTO. However, we do not recommend using DITTO due to the excessive JCL requirements. We recommend using TSIDDDTD and TSIDDDMP for equivalent functions each of which have minimal JCL requirements. When card-image processing is required, we recommend using your sort program (if that program supports card-image processing) or IDCAMS REPRO.</p>
DSN Catalog	<p>Only prime dataset definitions are allowed. Therefore, multi-dataset tape groups are specifically disallowed.</p>
JCL Requirements for Tape	<p>Each controlled dataset must be defined with a TLBL or DD statement.</p> <p>LUB allocation must be inhibited since DITTO requires use of the SYS number(s) specified in the DITTO control statement.</p> <p>Both input and output volumes must be allocated before DITTO is invoked by using the TSIDASS OPEN function and then repositioned back to load point.</p> <p>BIM-EPIC special option EPS022 must be enabled if BIM-EPIC reblocking options are invoked.</p> <p>BIM-EPIC special option EPS029 must be enabled which prevents BIM-EPIC from clearing the dataset name from input label statements.</p>
JCL Requirements for Disk	<p>Each controlled dataset must be defined with a DLBL or DD statement</p> <p>Both input and output datasets must be allocated before DITTO is invoked by using the TSIDASS OPEN function.</p>
Restrictions	<p>Only Tape File functions and Disk (also called Sequential or SAM) File functions can be used with BIM-EPIC controlled datasets. Basic Tape functions and Basic Disk functions cannot be used with BIM-EPIC since those functions do not use logical IOCS.</p>
Examples	<p>Many sample job streams for DITTO applications are supplied when BIM-EPIC is installed. Please see the <i>BIM-EPIC Installation and System Guide</i>, "BIM-EPIC Sample Jobstreams".</p>

VSE/POWER ACCOUNT FILE

Description	<p>You can use BIM-EPIC to control account tape and sequential disk files created by the VSE/POWER PACCOUNT command.</p>
DSN Catalog	<p>No tape dataset restrictions exist.</p> <p>For disk, the pool must be comprised of the exact same DASD type as is used for the VSE/POWER account file (IJAFIL). Specify an adequate primary allocation value as not all versions of VSE/POWER will request secondary allocations even if needed. Historically, a primary allocation value equal to the allocated size of IJAFIL has proven adequate.</p>
JCL Requirements for Tape	<p>Controlled account files must be defined with a TLBL statement stored in the system standard label area, VSE/POWER partition standard label area, or VSE/POWER partition user label area.</p> <p>LUB allocation must be inhibited since VSE/POWER will dynamically choose a SYS number which must be used.</p>
JCL Requirements for Disk	<p>Controlled account files must be defined with a DLBL statement stored in the system standard label area, VSE/POWER partition standard label area, or VSE/POWER partition user label area.</p> <p>If the controlled account file is inhibited for LUB allocation, supply an abbreviated EXTENT statement (// EXTENT SYSnnn) reflecting a SYS number that is otherwise not used by VSE/POWER.</p> <p>If the controlled account file is not inhibited for LUB allocation, supply an abbreviated EXTENT statement (// EXTENT SYSnnn) reflecting any valid SYS number.</p>

VSE/POWER PDISPLAY, POFFLOAD and Printer/Punch Tape Spooling

Description

With the introduction of standard label support for tape spooling of either print or punch data and for POFFLOAD tapes, it is possible to utilize BIM-EPIC controlled tapes and dataset names for these functions even though some restrictions do apply. To invoke BIM-EPIC control of these tape datasets, the TLBL= and LTAPE= operands need be specified in the appropriate VSE/POWER command or JECL control statement.

DSN Catalog

BIM-EPIC datasets must be defined to the BIM-EPIC catalog as prime datasets only. Do not attempt to use NSU (subordinate) datasets. With NSU datasets, VSE/POWER may successfully write the data but may be unable to read it because additional tape-marks are typically imbedded within that data.

Datasets may be cataloged as unlabeled. A better choice might be to only use labeled datasets since VSE/POWER will not be able to recognize unlabeled multi-volume versions beyond the first volume. It becomes the users responsibility to recognize that unlabeled multiple volumes exist within a version so the remaining volumes can be processed with additional VSE/POWER commands.

JCL Requirements

The actual TLBL control statement referenced by the TLBL= operand MUST be stored in the standard label area, the VSE/POWER partition label area, or the VSE/POWER user label area. LUB allocation must be inhibited since VSE/POWER will dynamically choose a SYS number which must be used.

VSE/POWER continues to make and release tape device assignments. This prevents BIM-EPIC from accomplishing AVR. VSE/POWER determines which tape drive to be used by the specification supplied in the PDISPLAY or POFFLOAD command, in the TADDR operand on the * \$\$ LST or * \$\$ PUN JECL statements, or, if the TADDR operand is not specified, by the System console response given to message 1Q55D SPECIFY TAPE ADDRESS.

If output fills more than a single tape volume, VSE/POWER will invoke VSE/SAM End-of-Volume procedures, unload the current tape and issue the message

```
4140A NO ALTERNATE DRIVE ASSIGNED ... SYSXXX=CUU
```

Since VSE/POWER still has the tape drive assigned, BIM-EPIC is unable to AVR other tape drives for an available scratch tape. The operator must mount a fresh scratch tape on the indicated drive and respond 'NEWTAP'.

If input spans multiple tape volumes, VSE/POWER will invoke VSE/SAM End-of-Volume procedures, unload the current tape and issue the message:

```
4140A NO ALTERNATE DRIVE ASSIGNED ... SYSXXX=CUU
```

Since VSE/POWER still has the tape drive assigned, BIM-EPIC is unable to AVR other tape drives for the next volume in the version. The operator must mount the required volume on the indicated drive and respond 'NEWTAP'.

SQL/VSE Tape Output Functions

Description

SQL/VSE is a long running task that manages database files for other tasks in the same VSE environment. The ability to archive database files (and their corresponding journal files) without having to terminate the SQL/VSE task is accomplished by invoking the operator communications facility for the partition-id in which the SQL/VSE task is running (MSG pp, where pp is the two character partition-id desired). At that point, SQL/VSE asks the console operator various questions. However, accomplishing archive functions of SQL/VSE where the output tape(s) are under the control of BIM-EPIC requires special considerations.

The problem arises because both BIM-EPIC and SQL/VSE want to manage the physical device allocation processes. Such processes are expected as part of a dataset management product such as BIM-EPIC but unusual when found as part of a database management product such as SQL/VSE. When both products are present in the same VSE environment, a conflict arises. Attempts to totally eliminate all the conflicts has not been entirely successful. However, the following considerations will allow you to accomplish your necessary SQL/VSE tape output operations and still maintain BIM-EPIC control of those tapes.

Although this discussion relates to the SQL/VSE tape output function for ARCHIVE, the same concepts covered by this discussion should apply to other tape output operations.

Configuration Requirements

Your BIM-EPIC system *must* be configured with TAPE=YES and TAPLUB=YES (or *nnn*).

DSN Catalog

Each tape dataset name placed in the TLBL control statement for use with the ARCHIVE function must be cataloged and the SYSnnn operand specified with a value other than SYS005. The logical unit number chosen must be outside the range of numbers that might be used for other SQL/VSE requirements. Only prime dataset names should be used and the definition for each unique dataset name used may reflect the same logical unit number for the SYSnnn operand.

JCL Requirements

The TLBL's required for each output file should reflect an option value of '2' to assure each output tape volume is unloaded when closed.

Once the ARCHIVE request has been entered, SQL/VSE needs to know if you wish to use a particular tape device which has been previously allocated to the SQL/VSE partition (called STATIC by SQL/VSE) or any currently available tape device of your choice (called DYNAMIC by SQL/VSE). Respond with the cuu of any currently available tape device and note that this particular tape device will not actually be used for any purpose by SQL/VSE and will also not be available to any other task for the duration of the archive process. This is an unavoidable requirement. Each output tape dataset created during the archive process is written to tape residing on a tape device other than the one specified above. Therefore, the archive process always requires the availability of two tape devices at a minimum.

VSE/ICCF and Interactive Partition GETVIS Requirements with BIM-EPIC

Problem

An ICCF pseudo partition is allocated only 48K of GETVIS and there is currently no way to change that default. With BIM-EPIC active in the VSE system (it makes *no* difference if BIM-EPIC is turned off in the ICCF/CICS partition using `OPTION EPICOFF=pp`), more than 48K of GETVIS is generally needed just to initiate many programs. The DTSUTIL program nearly always needs more than 48K of GETVIS and if sufficient GETVIS is not available, abnormal termination due to various program exception conditions (i.e., addressing, protection segment-translation, or operation) or BIM-EPIC messages EP042 and EP042A (both referring to an insufficient GETVIS condition) may result. Invoking DTSUTIL by either of the following commands entered on the command line will result in the default of 48K of GETVIS being allocated and will probably result in one of these failures:

```
$DTSUTIL
/RUN DTSUTIL
```

In addition, if the DTSUTIL program is invoked from the VSE/Interactive Interface, a CDLOAD failure could occur. There is no relationship between the amount of GETVIS that has been allocated to the ICCF/CICS partition itself - this discussion pertains to the pseudo partitions sub-allocated by ICCF and managed by the ICCF dispatcher.

Circumvention

This problem can be circumvented for the DTSUTIL program by storing a member named DTSUTIL in the VSE/ICCF library file (preferably in the common library) which contains the following two lines

```
/LOAD DTSUTIL
/OPTION GETVIS=256K
```

(DTSUTIL as a member name is used here only as a convenience - any name acceptable to ICCF could have been used.) The /LOAD job control statement must always be first. Notice the /OPTION job control statement which has the GETVIS operand specified. By specifying a particular GETVIS value, we can temporarily override the default GETVIS allocation of 48K and provide an adequate GETVIS storage for the DTSUTIL program. 256K of GETVIS is probably too much but a smaller value has not been tested for adequacy.

Once this member has been stored, enter the following command on the command line when it is desired to run the DTSUTIL program:

```
/EXEC DTSUTIL
```

By issuing this command, VSE/ICCF will search for the first library file member name of DTSUTIL and execute the ICCF job control found in that library member. Now, with adequate GETVIS storage provided, DTSUTIL will continue processing successfully.

Processing with COBOL II

Description

BIM-EPIC requires that COBOL II programs are compiled with the Library Management Feature (the RES CBL option) in order to function properly.

In some instances, you may have COBOL II software which was not compiled with the Library Management Feature active, and you have no access to the source code. To run these programs with BIM-EPIC, special JCL is required.

DSN Catalog

No special entries are required.

JCL Requirements

For all controlled disk datasets: Pre-open each dataset with the TSIDASS OPEN function.

For all uncontrolled disk datasets: The proper volume serial number must be specified on the EXTENT statement, and you must make an assignment to the proper device with a VSE ASSGN statement.

For all tape datasets: Assign the dataset to a tape drive with a VSE ASSGN statement or with the TSIDASS ASSGN function.

Examples

The following example processes a controlled input disk dataset. The dataset is pre-opened with TSIDASS.

```
// DLBL DISKIN, 'TEST.INPUT.DISK, (L) '
// EXEC TSIDASS
   OPEN DISKIN,SYS020,INPUT
/*
// EXEC PROGA
```

The following example processes an uncontrolled input disk dataset. VSE ASSGN and EXTENT statements are used to satisfy the COBOL II pre-open process.

```
// ASSGN SYS020,154
// DLBL DISKIN,'TEST.INPUT.DISK'
// EXTENT SYS020,VOL154,1,0,15,75
// EXEC PROGA
```

The following example processes a controlled tape dataset. The TSIDASS ASSGN function is used to satisfy the COBOL II pre-open process.

```
// DLBL TAPEIN,'TEST.TAPE',,,,,,64
// EXEC TSIDASS
   ASSGN SYS011
/*
// EXEC PROGA
```

Using BIM-EPIC's JCL Features

Restarting Programs from a Checkpoint

Description

To prepare to restart programs from a checkpoint:

- Close all output datasets with the TSIDMNT CLOSE function.
- Use the console log to determine which input and output tape volumes have been used and which checkpoints have been taken.

DSN Catalog

All datasets must be cataloged. Start Track 1 datasets cannot be used with checkpoint/restart. You must specify the SYS number in the catalog entry for disk datasets. You can also specify it for tape datasets.

JCL Requirements

To prepare the restart job:

1. Ensure that you use the same job name, supervisor, and partition.
2. Execute the TSIDASS RSTRT function in the step just prior to the VSE RESTRT command. See "Pre-Open Processing" for details.
3. Define each controlled dataset in the TSIDASS RSTRT step. Remember that the output datasets are now the current versions in the DSN Catalog, so submit all output RSTRT statements first, input statements last.

For tape, specify the volume sequence number on the label, if you restart with volume 2 or later. If you are processing a multi-dataset group, BIM-EPIC positions the tape automatically. Inhibit LUB allocation.

For disk, supply the EXT parameter reflecting the last extent opened.

4. Follow the TSIDASS step with a VSE ASSGN statement for the checkpoint dataset. Use the logical unit number that was used when the dataset was opened in the previous execution. See the EP039 and EP053 messages to find these numbers.
5. The SYS number used for the checkpoint dataset must be assigned with a VSE ASSGN statement if checkpoints are not embedded in the output dataset.
6. Execute the restart job.

Example

In this example, TSIDASS restarts both input and output datasets at volume 2, using the same SYS numbers that were used when the checkpoint was taken. The TLBL option "64" inhibits LUB allocation. Because the checkpoints are embedded (being written to the same output tape as the output dataset using an assignment of SYS019), a VSE ASSIGN statement is used to assign SYS019 to the checkpoint dataset. The job is restarted at checkpoint 3.

```
// JOB TESTCHKP
// TLBL TAPIN,'TEST.FILE1',,,2,,,64
// TLBL TAPOUT,'TEST.FILE2',,,2,,,64
// EXEC TSIDASS
RSTRT TAPOUT,SYS021,OUTPUT
RSTRT TAPIN,SYS020,INPUT
/*
// ASSIGN SYS019,SYS021
// RSTRT SYS019,3
/&
```

Choosing the Device Type

Generic Device Independence

Generic Device Independence (GDI) and Generic Device Independence for Unit Record Devices (GDIUR), sometimes called storage media independence, allows you to choose the type of media on which your dataset will be stored. You can process the dataset on either tape and disk, no matter how the dataset is defined in your program.

In addition, Generic Device Independence for Unit Record devices (GDIUR), allows you to use disk or tape for datasets that have been defined in your programs as unit record devices (DTF types DTFCF, DTFPR, DTFCP, and DTFDI). GDIUR can be used for logical units SYS000 through SYS255, and for SYSIPT, SYSLST, and SYSPCH.

GDIUR greatly simplifies the use of disk for VSE datasets that use the logical units SYSIPT, SYSLST, and SYSPCH. The following standard VSE JCL could be used to create a disk dataset on logical unit SYSPCH, and then read that dataset with logical unit SYSIPT:

```
// DLBL IJSYSPH, 'COBOL.PROGRAM', 0
// EXTENT SYSPCH, VOL251, 1, 0, 200, 2000
   ASSGN SYSPCH, DISK, VOL=VOL251, SHR
// EXEC DFHECP1$
.
.
   CLOSE SYSPCH, PUNCH
// DLBL IJSYSIN, 'COBOL.PROGRAM'
// EXTENT SYSIPT, VOL251
   ASSGN SYSIPT, DISK, VOL=VOL251, SHR
// EXEC FCOBOL
.
.
   CLOSE SYSIPT, SYSRDR
```

There are two problems with this approach:

- If the first step (DFHECP1\$) cancels, SYSPCH will remain permanently assigned to disk. Manual intervention or additional JCL is needed to re-assign SYSPCH correctly.
- If the second step (FCOBOL) cancels, SYSIPT remains permanently assigned to disk. Manual intervention or additional JCL is needed to re-assign SYSIPT correctly.

In addition, the JCL required is complex and lengthy, especially when compared to the following BIM-EPIC JCL which accomplishes the same task:

```
* //IJSYSPH DD DSN=$$$$.COBOL, UNIT=DISK, DISP=( , KEEP, DELETE )

// EXEC DFHECP1$

.

.

* //IJSYSIN DD DSN=$$$$.COBOL, DISP=( OLD, DELETE )
```

Note that no ASSGN or CLOSE statements are needed, because GDIUR ensures that system logical units can never be left erroneously assigned to disk.

Programs that use GDIUR can also take advantage of BIM-EPIC's reblocking facility to save disk space and improve performance.

Eligibility

You can use GDI or GDIUR for uncontrolled and controlled datasets. A dataset normally processed as a Start Track 1 disk dataset would become uncontrolled if processed on tape with GDI.

GDIUR does not support RPG-II SYSLST processing.

GDIUR datasets cannot have secondary extents.

GDI applies to tape and disk sequential datasets (DTFMT or DTFSD). VSAM, direct access, and ISAM datasets are not eligible.

GDI and GDIUR datasets must be processed with logical IOCS.

Configuration Options

Generic device independence is controlled by the configuration options GDI and GDIUR.

Processing

Device independence can be invoked in any of the following ways:

- In response to a message. An insufficient space message or a request for a scratch tape can be answered with TAPE or DISK to force the use of a different device type. (GDI only)
 - Automatically for input datasets. BIM-EPIC allocates the correct device type when controlled datasets are processed as input. (GDI only)
 - Using JCL. The desired device type is indicated by your job control statements. For example, use a TLBL instead of a DLBL to specify that the dataset is to be on tape instead of disk. On a DD statement, specify UNIT=DISK instead of UNIT=TAPE to specify that the dataset is to be on disk instead of tape. (Both GDI and GDIUR)
-

DSN Catalog

Required information for tape and disk must be supplied in the dataset definition for controlled datasets. This includes pooling and allocation information. If it is not included, BIM-EPIC prompts the operator for the necessary input.

JCL Requirements

For uncontrolled tape or disk datasets, use the proper label (TLBL, DLBL, or a DD statement with the UNIT parameter).

For controlled datasets, no JCL changes are required for input (GDI only). For output GDI, and GDIUR input and output, invoke GDI by changing the label type (TLBL or DLBL) or device type (UNIT=type). No volume or allocation information needs to be supplied in the JCL.

When using GDIUR, it may be necessary to insert the OPTION=RESET parameter on the DD statement after the step to terminate GDIUR processing if subsequent steps reference the same system dataset DTF name and you don't need to control it further using GDIUR. See 6-27 in this manual for more information.

Example

This example routes punch output to a disk dataset using GDIUR.

```
* //IJSYSPH DD DSN='CARD.OUTPUT',DISP=(NEW,PASS),
* //          UNIT=DISK
// EXEC PROGRAM
/*
* //IJSYSPH DD OPTION=RESET
```

System Datasets and VSE Device Independence

Description

Although we recommend that you use GDIUR, you can also control a system dataset that is routed to disk or tape using VSE device independence.

DSN Catalog

No special entries are required.

JCL Requirements

The dataset cannot have secondary extents.

The dataset must be permanently assigned, and label information must precede the assignment.

BIM-EPIC assigns VSE system datasets to the correct disk volume, so the assignment can specify any volume in the disk pool.

Example

This example routes SYSPCH output to a BIM-EPIC controlled work dataset, which is then processed using SYSIPT in the subsequent step.

```
// JOB ASSEMBLE AND CATALOG AN OBJECT MODULE
// DLBL IJSYSPH, '==.WORKFILE.1'
ASSGN SYSPCH, DISK, VOL=SYSWK1, SHR
// EXEC ASSEMBLY
   (SOURCE CODE)
/*
CLOSE SYSPCH, PUNCH
// DLBL IJSYSIN, '==.WORKFILE.1'
ASSGN SYSIPT, DISK, VOL=SYSWK1, SHR
// EXEC LIBR
CLOSE SYSIPT, READER
/&
```

Dummy Datasets and ASSGN IGNORE

Description	Dummy processing and ASSIGN IGNORE are only supported for LIOCS sequential access methods.
--------------------	--

DSN Catalog	No special entries are required.
--------------------	----------------------------------

JCL Requirements	<p>Assign the logical unit to IGNORE, so that BIM-EPIC ignores the dataset.</p> <p>Use an EXTENT statement for disk datasets defined with DLBLs.</p> <p>For DD statements, specify the DUMMY parameter instead of the DSN parameter.</p>
-------------------------	--

Examples	<pre>// ASSGN SYS020,IGN // TLBL TAPEOUT,'DATA.SET.A' // ASSGN SYS004,IGN // DLBL DISKOUT,'DATA.SET.B(L) ' // EXTENT SYS004 * //TAPEOUT DD DUMMY</pre>
-----------------	--

Concatenating Datasets

Description

Dataset concatenation reads multiple versions of a dataset or multiple datasets with one open. This allows you to process a number of datasets as if they were a single dataset.

All versions of concatenated datasets must reside on the same device type. If the datasets have different block sizes, you must read the dataset with the largest block size first.

DSN Catalog

No special entries are required.

JCL Requirements

With TLBL or DLBL statements: Use a separate TLBL or DLBL statement for each dataset to be read. Datasets are processed in the order in which their label statements appear. Use the same DTF name for each statement and supply the concatenation option code for all except the last version: TLBL option "48" and DLBL option "E".

WARNING!

Do not specify the concatenation code on the last label statement in a concatenation group!

Examples

The following example processes versions 1 through 5 of the DAILY.TRANS dataset on disk with one open. It also processes WEEKLY.TRANS and WEEKLY.RERUN on tape with another single open. Although there are mixed device types in the job, they are not mixed within a single open, so they meet the requirements for concatenation.

```
// DLBL INPUT1, 'DAILY.TRANS, (E,V=5) '
// DLBL INPUT1, 'DAILY.TRANS, (E,V=4) '
// DLBL INPUT1, 'DAILY.TRANS, (E,V=3) '
// DLBL INPUT1, 'DAILY.TRANS, (E,V=2) '
// DLBL INPUT1, 'DAILY.TRANS '
// TLBL INPUT2, 'WEEKLY.TRANS' , , , , , 48
// TLBL INPUT2, 'WEEKLY.RERUN'
// EXEC PROGRAM
```

With DD statements: Concatenation is automatic for all versions of a single dataset on input. Simply specify DSN=data.set.name without specifying a particular version. When all versions of a single dataset are concatenated, datasets are read in order from the most current version to the oldest.

In the following example, all versions of DAILY.TRANS are read with one open. For input with different dataset names, request concatenation explicitly. Specify the version number of each version to be read in the DSN parameter. Specify a ddname for the first statement only.

```
* //INPUT1 DD DSN='DAILY.TRANS',DISP=(OLD,KEEP)
```

The following example concatenates versions 1 through 4 of the DAILY.TRANS dataset and version 1 of the DAILY.DEP dataset.

```
* //INPUT1 DD DSN='DAILY.TRANS(3)',DISP=(OLD,KEEP)
* // DD DSN='DAILY.TRANS(2)',DISP=(OLD,KEEP)
* // DD DSN='DAILY.TRANS(1)',DISP=(OLD,KEEP)
* // DD DSN='DAILY.TRANS(0)',DISP=(OLD,KEEP)
* // DD DSN='DAILY.DEP(0)',DISP=(OLD,KEEP)
// EXEC PROGRAM
```

Processing Multiple Datasets with the Same DTF Name (Dataset Drop)

Description

BIM-EPIC allows you to open multiple datasets with the same DTF name. Each dataset is opened and closed separately, in the order in which its label is submitted.

DSN Catalog

No special entries are required.

JCL Requirements

Specify the dataset drop option for all but the last dataset defined.

For...	Use...
DLBLs	T
TLBLs	16 or D
DD statements	Automatic. Just use multiple DD statements with the same ddname. <i>This technique is not compatible with MVS.</i>

Examples

Dataset drop is invoked for three datasets using the ddname TAPEIN.

```
* //TAPEIN DD DSN=PAYROLL.WEEKLY(2),DISP=(OLD,KEEP)
* //TAPEIN DD DSN=PAYROLL.WEEKLY(1),DISP=(OLD,KEEP)
* //TAPEIN DD DSN=PAYROLL.WEEKLY(0),DISP=(OLD,KEEP)
```

Dataset drop is invoked for three datasets using the DTF name TAPEIN. Option 16 is used for all but the last dataset with that DTF name.

```
// TLBL TAPEIN,'FILE1',,,,,,16
// TLBL TAPEIN,'FILE2',,,,,,16
// TLBL TAPEIN,'FILE3'
```

Running Test Jobs With Production Datasets

Description

BIM-EPIC’s test job feature allows you to run test jobs with production datasets without affecting production processing. Test job names are defined by the TSTJOB configuration option. The TSTJOB configuration option can include wildcard characters to allow greater flexibility in creating test job names. A question mark (?) specifies that any character in that position is a valid character. An asterisk (*) specifies that all characters in that position and any following are valid.

All versions of controlled datasets created using test jobs are recorded in the DSN Catalog with a status of "conditionally cataloged". Only such versions can be accessed by subsequent test job executions. Production jobs will not use the conditionally cataloged versions. These versions are retained until the following day, when they become eligible for scratch. If you close a conditionally cataloged version using Online Manager (or TSIDMNT) during that time, it becomes an active version (version 1), can be accessed by production jobs, and is maintained according to its EDD retention criteria.

DSN Catalog

No special entries are required.

JCL Requirements

To test a single dataset, specify a conditional catalog option on its label:

For...	Use...
DLBLs	C
TLBLs	32
DD statements	DISP=(xxx,CONDCAT)

To test an entire job, specify a valid test job name on the VSE JOB statement.

Examples

If TSTJOB=A*, the test job feature is activated by any job name beginning with A.

If TSTJOB=A?AAAA, the test job feature is activated by the job names ABAAAA, AXAAAA, A1AAAA, and so on.

If TSTJOB=A?A*, the test job feature is activated by the job names ABAAAA, AXA, ACAXX, and so on.

8

BIM-EPIC Utilities

This chapter describes how to use BIM-EPIC JCL utilities that control files that are processed with physical IOCS (PIOCS) instead of logical IOCS (LIOCS). For more information about the recommended use of these utilities, see 7-1 in this manual. In addition, this chapter presents the dataset copy and dump utilities TSIDDTD and TSIDDMF and the VSE label access subroutines (TSIDGLB and TSIDPLB).

Chapter Contents

Pre-Open Processing (TSIDASS)	8-2
Deleting Datasets (TSIDDEL)	8-7
Printing Datasets (TSIDDMF)	8-16

Pre-Open Processing (TSIDASS)

Description

Use TSIDASS to process controlled files with programs that do not issue OPEN requests or that do pre-open checking of assignments. You can also use TSIDASS to make assignments to SYSRDR, SYSIPT, SYSLST, and SYSPCH.

Its functions are determined by control statements:

- ASSGN satisfies pre-open assignment checking for tape datasets
- OPEN opens datasets for programs which do not issue an OPEN request
- RSTRT assists with using IBM checkpoint restart facility

TSIDASS is executed immediately before the program that does pre-open assignment checking. Inhibit LUB allocation for all datasets in the TSIDASS step by using the appropriate option codes on the TLBL and DLBL statements in the step.

Note

Pre-open assignment checking can also be satisfied with the DD parameter OPTION=SYSnnn.

JCL Requirements

```
dataset labels
// EXEC TSIDASS
control statements
/*
// EXEC <program that performs pre-open assignment checking>
```

UPSI Values

None

Dataset Requirements

Place the labels for all datasets *before* the EXEC statement for TSIDASS. Do not place any labels between the EXEC statement for TSIDASS and the EXEC statement for your program.

ASSGN Statements

Use the ASSGN function for tape datasets only. It satisfies pre-open assignment checking by assigning the logical unit to a dummy tape device. When the program issues an OPEN request for the dataset, BIM-EPIC performs its usual device assignment to the real device.

For sort programs, we recommend using TSIDSRT to perform this function instead of TSIDASS. See 7-37 in this manual for more information and JCL examples.

You can also use ASSGN to make assignments to the system logical units SYSRDR, SYSIPT, SYSLST, and SYSPCH. When you make these assignments, be sure to submit their control statements last, because TSIDASS stops processing control statements after one of these assignments is made.

To make the assignment to a system logical unit, you must supply a dataset label for the *dtfname* IJSYSIN, IJSYSLS, or IJSYSPH. If a SYSPCH or SYSLST dataset is created on tape, you must place it in closed status before you can use it as input. Use the CLOSE function of TSIDMNT to close the dataset after it is created.

To simplify the JCL necessary to process system files (SYSIPT, SYSLIST and SYSPCH), use the Generic Device Independence for Unit Record Devices (GDIUR) feature. The GDIUR feature eliminates the need for extra TSIDASS execution, VSE CLOSE statements and most of the TSIDMNT CLOSE function executions. GDIUR also protects your default system file assignments in case the job terminates abnormally. Refer to "Choosing the Device Type" in Chapter 7, "Running Jobs With BIM-EPIC" for additional information about using GDIUR.

Syntax

```
[//] ASSGN SYSxxx [,DUMMY]
```

Example

```
// JOB MAKE ASSIGNMENT WITH TSIDASS
// TLBL TAPIN1,'DAILY.TRANS',,,,,,64
// TLBL TAPOUT,'DAILY.BILLING',,,,,,64
// EXEC TSIDASS
  ASSGN SYS021
  ASSGN SYS022
/*
// EXEC BILLING
/&
```

In this example, the program BILLING uses SYS021 for TAPIN1 and SYS022 for TAPOUT. TSIDASS makes the assignment for both of these files before the BILLING program is executed. The TLBL option code 64 inhibits LUB allocation.

OPEN Statements

Use TSIDASS OPEN when you process controlled datasets with a program that does not issue OPEN requests. TSIDASS OPEN issues a logical open for the dataset, so that BIM-EPIC can gain control of the dataset.

Syntax

```
OPEN dtfname,SYSxxx,type
```

Parameters

Parameter	Required?	Valid Entry	Default
<i>dtfname</i>	Yes	1 to 7 characters	None
<i>xxx</i>	Yes	000-254	None
<i>type</i>	Yes	INPUT or OUTPUT	None

dtfname identifies the dataset to be opened. Specify the *dtfname* or *ddname* as it appears on the dataset's label statement.

SYSxxx specifies the logical unit to assign.

INPUT specifies that the file is opened as input.

OUTPUT specifies that the file is opened as output.

Example

```
// JOB OPEN TAPE DATASET
// TLBL TAPEIN, 'AP.DAILY.TRANS' , , , , , 64
// EXEC TSIDASS
  OPEN TAPEIN, SYS020, OUTPUT
/*
// EXEC PROGRAM
/ &
```

In this example, TSIDASS issues an open for the AP.DAILY.TRANS dataset. LUB allocation is inhibited for that dataset.

RSTRT Statements

Use TSIDASS RSTRT when you are restarting a job that has been checkpointed by VSE. Like OPEN, it issues a logical open for named datasets, allowing BIM-EPIC to maintain control of its datasets. It also allows BIM-EPIC to do AVR processing and assign tape drives. See 7-54 in this manual for detailed restart procedures and for a JCL example.

LUB allocation must be inhibited for all restarted datasets. In addition, the same SYS number that was in use when the checkpoint was taken must be specified.

Syntax

```
RSTRT dtfname, SYSxxx { , INPUT
                        , OUTPUT } [ , EXT=nn]
```

RSTRT Parameters

Parameter	Required?	Valid	Default
<i>dtfname</i>	Yes	1 to 7 characters	None
<i>xxx</i>	Yes	000-254	None
Specify one of the following two (required):			
INPUT		None	OFF
OUTPUT		None	OFF
EXT	No	1 to 255	1

dtfname identifies a dataset that will be used when the program is restarted. Specify the DTF name or DD name as it appears on the dataset's label statement.

SYSxxx specifies the logical unit to assign. It must be the same logical unit number that was in use when the checkpoint was taken.

INPUT specifies that the dataset was last opened as input.

OUTPUT specifies that the dataset was last opened as output.

EXT specifies the sequence number of the extent to be re-opened. This is only necessary for disk datasets, and only if the dataset is to be opened at an extent other than the primary extent. This number will be equal to the number of BIM-EPIC EP053 messages that were issued for the dataset prior to the checkpoint from which the program will be restarted.

Using the BIM-EPIC Sort Interface (TSIDSRT)

Description	<p>TSIDSRT solves the problems that can occur at both open and close with some SORT utility programs. TSIDSRT is not a sort program. Instead, it is an interface program between BIM-EPIC and your sort program. It calls your sort program to do the sort, but it allows BIM-EPIC to satisfy pre-open assignment checking, to truncate the SORTOUT file, and to close and delete SORTWK areas. TSIDSRT provides full support for BIM-EPIC's generic device independence and reblocking features.</p> <p>If you use unlabeled tape datasets in your sort, these datasets must be controlled.</p> <p>You can use only one SORTOUT dataset.</p> <p>For other specific information on executing a sort with controlled datasets and for JCL examples, see 7-37 in this manual.</p>
Configuration Options	<p>GDI=YES must be specified as a configuration option.</p>
JCL Requirements	<pre>// <labels for sort datasets> // EXEC TSIDSRT sort control statements /*</pre>
UPSI Values	<p>None</p>
Dataset Requirements	<p>Supply labels for the datasets required by your sort program.</p>
Control Statements	<p>Supply the control statements required for your sort program.</p>
Example	<pre>// JOB EXECUTE SORT // TLBL SORTIN1,'DAILY.TRANS',,,,,,2 // TLBL SORTOUT,'SORTED.DAILY.TRANS',,,,,,1 // DLBL SORTWK1,'\$\$\$.SORTWK1,(D) ' // EXEC TSIDSRT SORT FIELDS=(33,4,A),FORMAT=BI,WORK=1,FILES=1 RECORD TYPE=F,LENGTH=80 INPFIL BLKSIZE=1600 OUTFIL BLKSIZE=4000 END /* /&</pre>

Deleting Datasets (TSIDDEL)

Description

Some programs that use PIOCS do not close their work datasets. TSIDDEL can be used to delete disk work datasets that are not closed by the program that opens them. At any point in the job or job step, TSIDDEL can be used to free up their disk space for other processing. These datasets must be controlled, and they can be either Start Track 1 or cataloged datasets.

TSIDDEL is available as a batch program and as a callable subroutine. When using a DLBL, specify the DLBL 'D' option. When using a BIM-EPIC DD statement, specify DISP=(NEW,DELETE).

If you are already using TSIDSRT as the interface to your sort program, TSIDDEL execution is not necessary.

TSIDDEL does not delete generation disk datasets.

JCL Requirements

```
// EXEC TSIDDEL
[ control statements ]
/*
```

UPSI Values

None

Dataset Requirements

The datasets must have been created in the step immediately prior to the TSIDDEL step, with no labels between the EXEC statement for the program and the EXEC statement for TSIDDEL.

Control Statements

Control statements specify which datasets to delete. Each control statement specifies one dataset to delete, but you can submit as many control statements as you wish for each execution of TSIDDEL. Control statements must begin in column 1.

Syntax

dtfname

TSIDDEL Parameter

Parameter	Required?	Valid	Default
<i>dtfname</i>	Yes	1 to 7 characters	None

dtfname specifies the DTF name or DD name of the dataset to be deleted.

Example

The following job creates the SORTWK1 dataset in the first step. The 'D' option is included on the DLBL to qualify the dataset for deletion. In the second step, TSIDDEL deletes the SORTWK1 file and frees the space it occupied.

```
.
.
.
// DLBL SORTWK1,'$$$'.SORTWK1,(D)'
// EXTENT SYS003
// EXEC SORT
[sort control statements]
/*
// EXEC TSIDDEL
SORTWK1
/*
.
.
.
```

Using TSIDDEL as a Subroutine

When you use TSIDDEL as a subroutine, you must use a parameter list to specify which file to delete. In the parameter list, place the 1 to 7-character DTF name or DD name as it appears on the dataset's label statement. You can delete only one file per call.

Control is passed to the subroutine using standard IBM linkage conventions. TSIDDEL does not provide any return codes or issue any messages.

Assembler Example

```
...      CALL TSIDDEL,(FILEID)
...
FILEID DC    CL7'SORTWK1'
...
```

COBOL Example

```
...
WORKING-STORAGE SECTION.
10 FILEID PIC X(7) VALUE 'SORTWK1'.
...
PROCEDURE DIVISION.
...
CALL 'TSIDDEL' USING FILEID.
...
```

Copying and Merging Datasets (TSIDDDTD)

Description

TSIDDDTD is a dataset-to-dataset copy/merge utility program. It can be used to copy a single dataset or to merge multiple input datasets into a single output dataset. TSIDDDTD uses standard LIOCS access methods, so the input and output datasets can be completely controlled by BIM-EPIC.

TSIDDDTD supports any sequential dataset with a fixed, variable, spanned or undefined record format. However, TSIDDDTD can only process tape datasets that follow the standard LIOCS format conventions (see "Tape Formats Supported" below).

TSIDDDTD performs one copy operation per execution. To copy a multi-file tape, execute TSIDDDTD once for every dataset on the tape, and include input and output labels for each dataset.

Configuration Options

To use TSIDDDTD for disk datasets, GDI=YES must be specified.

JCL Requirements

```
input dataset label(s)
output dataset label
// EXEC TSIDDDTD
[optional control statements ]
/*
```

UPSI Values

None

Dataset Requirements

You must submit a label for at least one input dataset and one output dataset. The default DTF names are INPUT and OUTPUT, but you can specify other DTF names in your control statements.

The output dataset's block size will be the same as the input dataset's block size unless you specify a new block size in the output dataset's definition. Uncontrolled disk datasets can be reblocked using the 'B=n' option on the DLBL statement.

When you are copying uncontrolled datasets, use SYS005 for the output dataset and SYS004 for the input dataset.

If more than one input dataset is being copied to the output dataset, specify the 'T' option (dataset drop) on every input dataset label except the last. With the 'Drop' option, you can merge tape and disk versions into a single dataset. If all the input is on the same device type, you can also use the concatenation feature of the BIM-EPIC DD statement.

Tape Formats Supported

Because TSIDDDTD uses standard LIOCS access methods, it can only process tapes which follow standard LIOCS format conventions.

For a labeled tape, the format is:

VOL/HDR labels, tapemark, data, tapemark, EOVS/EOF labels, tapemark

For an unlabeled tape, the format is:

[tapemark], data, tapemark

Some utility programs which appear to use LIOCS when writing standard label tapes actually use PIOCS and imbed additional tape marks within the data area. Utilities known to do this include the IDCAMS BACKUP function (which is different from the REPRO function) and the LIBR BACKUP function. When LIOCS encounters one of these additional tape marks on input, it issues the message "4130D EOF OR EOVS INQUIRY?" because no EOVS/EOF label was found immediately following that tape mark. Therefore, should the 4130D message occur while copying a file, it indicates that the tape format does *not* follow standard LIOCS conventions and *cannot* be copied successfully with TSIDDDTD.

Unfortunately, there is no method currently available to copy tapes which use non-standard formats. Use of DITTO basic functions would not be a good choice because that would result in two (or more) volumes reflecting the same volume serial number. The best known solution is to execute the same program which created the original volume(s) a second time to create a second version.

Control Statements

Control statements are used to define dataset characteristics and specify TSIDDDTD processing options.

For controlled datasets, control statements are not normally necessary. TSIDDDTD will get the record type, record size and block size from the version record in the DSN Catalog.

For uncontrolled and Start Track 1 datasets, control statements specifying the record type, record size and block size are required.

The TSIDDDTD maximum block size for tape is 32K. Block sizes greater than 32K are supported for disk. For uncontrolled labeled input tapes, TSIDDDTD will use HDR2 information. If record type, record size and block size are not specified and cannot be obtained from the DSN Catalog, TSIDDDTD defaults to an undefined record format with a blocksize of 32767. Record type, record size, and block size may not be in the catalog for some controlled datasets that were created by programs that use PIOCS.

Control statement parameters can be used to specify alternate DTFnames, control unlabeled tape processing, limit total output, select specific records only and end processing before end-of-file (EOF) is reached on input.

All control statements will be read and processed before TSIDDDTD begins the copy operation. Parameters can be placed in columns 1 through 71. Separate parameters with commas. Multiple statements can be submitted; but parameters must be completed on the statement on which they begin.

Syntax

```

[ BLKSIZE=nnnnn ]
[ FEOF=start,length,operand,type,value ]
[ HDRINFO ]
[ INNAME=dtfname ]
[ INPUT=n ]
[ LRECL=nnnnn ]
[ NOREWIN ]
[ NOREWOUT ]
[ NOTPMK ]
[ OBLKSIZE=nnnnn ]
[ OUTNAME=dtfname ]
[ RECFM=x ]
[ RUNIN ]
[ RUNOUT ]
[ RECS=nnnnnn ]
[ SEL=start,length,operand,type,value ]
[ SKIP=nnnnnn ]
[ UNLBLIN ]
[ UNLBLOUT ]

```

Control Statement Parameters

Parameter	Required?	Valid	Default
BLKSIZE	No	1 to 32767 (or greater for disk)	32767
FEOF	No	See subparameters	None
FEOF Subparameters			
<i>start</i>		1 to 999999	None
<i>length</i>		1 to 999999	None
<i>operand</i>		EQ NE GT LT GE LE	None
<i>type</i>		C or X	C
<i>value</i>		See text	None
HDRINFO	No	None	OFF
INNAME	No	1 to 7 characters	INPUT
INPUT	No	'?' or 1 to 32767	None
LRECL	No	1 to 99999	None
OBLKSIZE	No	1 to 32767	BLKSIZE
OUTNAME	No	1 to 7 characters	OUTPUT
NOREWIN/NOREWOUT	No	None	OFF
NOTPMK	No	None	OFF
RECFM	No	F V S U	U

RECS	No	1 to 999999	All
RUNIN/RUNOUT	No	None	OFF
SEL	No	See FEOF	All
SKIP	No	1 to 999999	0
UNLBLIN/UNLBLOUT	No	None	OFF

BLKSIZE defines the block size for input datasets. If **BLKSIZE** is not specified, it will be set to one of the following:

- the value of **LRECL**
- undefined (if **LRECL** is not specified). In this case, the input dataset is treated as unblocked.

For controlled datasets, the version block size for input datasets and the EDD reblock size for output datasets are used instead if available.

FEOF specifies a condition that causes TSIDDDTD to act as if EOF had been reached on input. If multiple parameters are specified on a single control statement, only another FEOF parameter may follow an FEOF parameter. When any one of multiple FEOF conditions is met, EOF will be assumed. Up to a total of 20 FEOF parameters can be submitted per execution of TSIDDDTD.

When FEOF is specified, all FEOF subparameters are required and positional:

start specifies the starting position in the record (relative to 1) of the field to be tested.

length specifies the length of the field to be tested.

operand specifies the type of test.

Use...	To specify that the input data is...
EQ	equal to value
NE	not equal to value
GT	greater than value
LT	less than value
GE	greater than or equal to value
LE	less than or equal to value

type specifies the type of data to be used in the test. C indicates that the value is in character format. X indicates that the value is in hexadecimal format.

value specifies the constant to be used in the test. Specify the same number of characters as the length value for character comparisons, or two times that number for hexadecimal comparisons.

HDRINFO causes VSE to display the tape header information on the operator console when opening the input datasets.

INNAME specifies an alternate input DTF name.

INPUT specifies the number of input datasets to be merged into a single output. If specified as '?' (i.e., INPUT=?), the message EP712 will be issued requiring the operator to respond with the number of files to be copied.

LRECL defines the dataset record size. For catalog-controlled datasets, the LRECL in the DSN Catalog is used instead if available. If LRECL is not specified, it will be set to one of the following:

- the value of **BLKSIZE**
- undefined (if **BLKSIZE** is not specified). In this case, the input dataset is treated as unblocked.

NOREWIN prevents rewinding of input tapes at close.

NOREWOUT prevents rewinding of the output tape at close.

NOTPMK suppresses the leading tape marks normally written before the data on unlabeled output tapes.

OBLKSIZE defines the block size for the output dataset. If **OBLKSIZE** is not specified, the **BLKSIZE** value will be used.

OUTNAME specifies an alternate output DTF name.

RECFM specifies the record format of both input and output datasets.

Code	Record Format
F	fixed length
V	variable length
S	spanned
U	undefined

RUNIN unloads input tapes at close.

RUNOUT unloads the output tape at close.

RECS specifies the number of records to be copied.

SEL specifies selection criteria for copying. If the data matches the expression, it is copied. If it does not match, it is skipped. If multiple parameters are specified on a single control statement, only another SEL parameter may follow a SEL parameter. When any one of multiple SEL conditions is met, the data will be copied. Up to a total of 20 SEL parameters can be submitted per execution of TSIDDDTD. If no SEL parameters are submitted, all data is copied.

When SEL is specified, all SEL subparameters are required. SEL subparameters are identical to FEOF subparameters (see above).

SKIP specifies the number of records to skip before beginning the copy operation.

UNLBLIN defines the input as an unlabeled uncontrolled tape dataset (not required if the dataset is controlled).

UNLBLOUT defines the output as an unlabeled uncontrolled tape dataset (not required if the dataset is controlled).

Example

This job copies a controlled disk dataset to a controlled tape dataset. No TSIDDDTD control statements are required.

```
// JOB TSIDDDTD COPY DISK TO TAPE
// DLBL INPUT,'input.data.set'
// TLBL OUTPUT,'output.data.set'
// EXEC TSIDDDTD
/*
/ &
```

Example

This job merges two input disk datasets and two input tape datasets to a single output tape dataset. All of the datasets are controlled.

```
// JOB TSIDDDTD MERGE 2 DISK AND 2 TAPE DATASETS TO A SINGLE OUTPUT
// DLBL INPUT,'input.data.set.1,(T)'
// DLBL INPUT,'input.data.set.2,(T)'
// TLBL INPUT,'input.data.set.3',,,,,,18
// TLBL INPUT,'input.data.set.4',,,,,,2
// TLBL OUTPUT,'output.data.set'
// EXEC TSIDDDTD
INPUT=4
/*
/ &
```

Note the use of the 'Drop' option (T) on the DLBLs and Drop (16) and Rewind (2) options on the TLBLs.

Example

This job makes a duplicate copy of a multi-file tape containing three datasets. All of the datasets are controlled, so no control statements are necessary. TSIDDDTD is executed three times, once for each dataset. Note the use of the TLBL options 'Inhibit LUB allocation' (64), 'Hold assignment' (8) and 'No rewind' (3) on the first two TLBLs and then just 'Inhibit LUB allocation' (64) and 'Unload' (2) on the last TLBL.

```
// JOB TSIDDDTD COPY MULTI-FILE TAPE
// TLBL INPUT,'input.prime.dsn',,,,,,75
// TLBL OUTPUT,'output.prime.dsn',,,,,,75
// EXEC TSIDDDTD
/*
// TLBL INPUT,'input.sub.dsn.1',,,,,,75
// TLBL OUTPUT,'output.sub.dsn.1',,,,,,75
// EXEC TSIDDDTD
/*
// TLBL INPUT,'input.sub.dsn.2',,,,,,66
// TLBL OUTPUT,'output.sub.dsn.2',,,,,,66
// EXEC TSIDDDTD
/*
/ &
```

Example

This job copies a Start Track 1 disk dataset to an output dataset using alternate DTF names, selecting only those records with the constant "TAC" in positions 1-3 or any data except low-values in positions 4-6. The record format, record size and block size information is required for Start Track 1 datasets.

```
// JOB TSIDDDTD  CREATE NEWFIL
// DLBL OLDFIL,'input.data.set'
// EXTENT SYS005,POOL01,1,0,1,300
// TLBL NEWFIL,'output.data.set'
// EXEC TSIDDDTD
INNAME=OLDFIL,OUTNAME=NEWFIL
RECFM=F,BLKSIZE=4000,LRECL=80
SEL=1,3,EQ,C,TAC
SEL=4,3,NE,X,000000
/*
/&
```

Printing Datasets (TSIDDMP)

Description

TSIDDMP is a dataset dump and print utility program. It can be used to completely or selectively dump or print a dataset in hex or character format, and will also print report files that have been written to disk or tape. TSIDDMP uses standard LIOCS access methods, so the input dataset can easily be controlled by BIM-EPIC.

TSIDDMP supports any sequential dataset with a fixed, variable, spanned or undefined record format. However, TSIDDMP can only process tape datasets that follow the standard LIOCS format conventions (see “Tape Formats Supported” for TSIDDDTD above).

TSIDDMP performs one dump or print operation per execution.

Configuration Options

To use TSIDDMP for disk datasets, GDI=YES must be specified as a configuration option.

JCL Requirements

```
input dataset label
// EXEC TSIDDMP
[ control statements ]
/*
```

UPSI Values

None

Dataset Requirements

You must submit a label for the input dataset. The default DTF name is INPUT, but you can specify an alternate DTF name in your control statements.

For uncontrolled tape datasets, use SYS004 for the input dataset.

Control Statements

Control statements are used to define dataset characteristics and specify TSIDDMP processing options.

Control statements are not normally necessary for controlled datasets, because TSIDDMP will get the record type, record size and block size from the DSN Catalog.

Control statements specifying the record type, record size and block size are required for uncontrolled and Start Track 1 datasets.

If record type, record size and block size are not specified and cannot be obtained from the DSN Catalog, TSIDDMP will use an undefined record format with a blocksize of 32767.

(Note: record type, record size, and block size may not be available in the catalog for some catalog-controlled datasets if they were created by programs that use PIOCS..)

Control statement parameters can be used to limit total output, select specific records only and end processing before end-of-file (EOF) is reached on input. Various formatting options are also provided.

All control statements will be read and processed before TSIDDMP begins the dump operation. Parameters can be placed in columns 1 through 71. Separate parameters with commas. Multiple statements can be submitted; but parameters must be completed on the statement on which they begin.

Syntax

```
[ BLKSIZE=nnnnn ]
[ CHARONLY ]
[ CTLCHAR ]
[ FEOF=start,length,operand,type,value ]
[ HDRINFO ]
[ INNAME=dtfname ]
[ LRECL=nnnnn ]
[ NOREW ]
[ NOSCALE ]
[ RECFM=x ]
[ RECS=nnnnnn ]
[ RUN ]
[ SEL=start,length,operand,type,value ]
[ SKIP=nnnnnn ]
[ UNLBL ]
```

Parameters

Parameter	Required?	Valid	Default
BLKSIZE	No	1 to 99999	See text
CHARONLY	No	None	OFF
CTLCHAR	No	None	OFF
FEOF	No	See subparameters	None
FEOF Subparameters			
<i>start</i>		1 to 999999	None
<i>length</i>		1 to 999999	None
<i>operand</i>		EQ NE GT LT GE LE	None
<i>type</i>		C or X	C
<i>value</i>		See text	None
HDRINFO	No	None	OFF
INNAME	No	1 to 7 characters	INPUT
LRECL	No	1 to 32767	None
NOREW	No	None	OFF

NOSCALE	No	None	OFF
RECFM	No	F V S U	U
RECS	No	0 or 1 to 999999	All
RUN	No	None	OFF
SEL	No	See FEOF	All
SKIP	No	1 to 999999	0
UNLBL	No	None	OFF

Descriptions

BLKSIZE defines the block size for the input dataset. If BLKSIZE is not specified, it will be set to one of the following:

- the value of LRECL
- undefined (if LRECL is not specified). In this case, the input dataset is treated as unblocked.

For controlled datasets, the block size in the catalog will be used instead if available.

CHARONLY specifies that only the character representation is printed. If CHARONLY is omitted, both character and hexadecimal representations are printed.

CTLCHAR specifies that the dataset has an ASA print control character in the first byte of each record to control printer spacing. Use CTLCHAR for print files that have been written to disk or tape. CTLCHAR forces CHARONLY and NOSCALE printing.

NOTE: If CTLCHAR is used for a dataset that does not contain a valid ASA print character in the first byte of every record, TSIDDMP can cancel with an I/O error. To verify the print characters, print the file in hex mode. The first character should be one of the following: space(X'40'), 0-9, A-C, +, or -

FEOF specifies a condition that causes TSIDDMP to act as if EOF had been reached on input. If multiple parameters are specified on a single control statement, only another FEOF parameter may follow an FEOF parameter. When any one of multiple FEOF conditions is met, EOF will be assumed. Up to a total of 20 FEOF parameters can be submitted per execution of TSIDDMP.

When FEOF is specified, all FEOF subparameters are required and positional:

start specifies the starting position in the record (relative to 1) of the field to be tested.

length specifies the length of the field to be tested.

operand specifies the type of test.

Use...	To specify that the input data is...
EQ	equal to value
NE	not equal to value

GT	greater than value
LT	less than value
GE	greater than or equal to value
LE	less than or equal to value

type specifies the type of data to be used in the test. C indicates that the value is in character format. X indicates that the value is in hexadecimal format.

value specifies the constant to be used in the test. Specify the same number of characters as the length value for character comparisons, or two times that number for hexadecimal comparisons.

HDRINFO causes VSE to display the tape header information on the operator console when opening the input datasets.

INNAME specifies an alternate input DTF name.

LRECL defines the dataset record size. For catalog-controlled datasets, the LRECL in the DSN Catalog is used instead if available. If LRECL is not specified, it will be set to one of the following:

- the value of BLKSIZE
- undefined (if BLKSIZE is not specified). In this case, the input dataset is treated as unblocked.

NOREW Prevents rewind of the input tape at close.

NOSCALE Suppresses (printing of) the position scale, which normally appears under the print record.

RECFM specifies the record format of both input and output datasets.

Code	Record Format
F	fixed length
V	variable length
S	spanned
U	undefined

RECS specifies the number of records to be printed. If zero is specified, the input dataset version is opened and closed without attempting any read any records.

RUN Unloads input tape at close.

SEL specifies selection criteria for dumping and printing data. If the data matches the expression, it is dumped or printed. If it does not match, it is skipped. If multiple parameters are specified on a single control statement, only another SEL parameter may follow a SEL parameter. When any one of multiple SEL conditions is met, the data will be dumped or printed. Up to a total of 20 SEL parameters can be submitted per execution of TSIDDMP. If no SEL parameters are submitted, all data is dumped or printed.

When SEL is specified, all SEL subparameters are required. SEL subparameters are identical to FEOF subparameters (see above).

SKIP specifies the number of records to skip before beginning the copy operation.

UNLBL defines the input as an unlabeled uncontrolled tape dataset (not required if the dataset is controlled).

Examples

The following job will dump the entire contents of a catalog-controlled disk dataset in hex format. No control statements are required.

```
// JOB TSIDDMP  DUMP DISK DATASET IN HEX
// DLBL INPUT,'input.data.set'
// EXEC TSIDDMP
/*
/ &
```

The following will print the first 100 lines of a report file that has been routed to a controlled tape dataset by using GDIUR.

```
// JOB TSIDDMP  PRINT BLOCKED REPORT TAPE
// TLBL INPUT,'input.data.set'
// EXEC TSIDDMP
CTLCHAR,RECS=100
/*
/ &
```

The following job prints a start track one disk dataset using an alternate DTF name, in character-only format, with the position scale suppressed. Only records containing the characters 10794 in positions 1-5 will be selected. End of file (and end of job) will be forced as soon as high values are found in positions 7-9. Note that the record format, record size and block size information is required for an uncataloged disk dataset.

```
// JOB TSIDDMP  SELECTIVE DUMP OF S.T. 1 FILE
// DLBL TPDATA,'input.data.set'
// EXTENT SYS005,POOL01,1,0,1,300
// EXEC TSIDDMP
INNAME=TPDATA
RECFM=F,BLKSIZE=4000,LRECL=80
CHARONLY,NOSCALE
SEL=1,5,EQ,C,10794
FEOF=7,3,EQ,X,FFFFFF
/*
/ &
```


Accessing the VSE Label Area (TSIDGLB and TSIDPLB)

Description

TSIDGLB and TSIDPLB are subroutines that can be called by your program to read and write VSE labels. TSIDGLB is used to read system-standard, partition-standard, and partition user labels.

TSIDPLB is used to modify a label that was acquired by TSIDGLB. TSIDPLB writes the label to an area maintained by BIM-EPIC. When BIM-EPIC is active, it checks for the label in this area first before searching the VSE label area.

Using the Subroutines

You can call both subroutines at any time, before or after a dataset is opened or closed. Both are called using standard IBM linkage conventions.

You can read and modify the label of any dataset defined in your JCL. However, tape dataset names are restricted to 17 characters. If you want to create a new label, first read a TLBL or DLBL that has no parameters except the DTF name.

Sample assembler and COBOL programs (TSIGLBL.A and TSIGLBL.C) are provided as examples. These can be assembled or compiled and run to demonstrate the coding techniques required to process VSE labels. These programs can be found in the BIM-EPIC installation library.

Data Area

Use the assembler copybook (TSIDGLB.A), or the COBOL copybook (TSIDGLB.C), for both subroutines. The discussion below refers to the data names in the assembler copybook.

Specify FILENAME and LBLRETCD as parameter list entries on your CALL statement.

FILENAME contains the DTF name of the label to be processed. The same DTF name must be specified in JCL.

Both subroutines return a 1-byte binary value in LBLRETCD. A zero (0) indicates that the label was returned by TSIDGLB or written by TSIDPLB. Any non-zero value indicates the label was not found or was not written.

In LBLLEN, TSIDGLB returns a 2-byte, unaligned, binary field that contains the length of the label returned. A length of 80 bytes indicates the label is for tape. Any length greater than 80 indicates the label is for disk. TSIDPLB uses the label length from this field.

TSIDGLB returns the label beginning at LBLFILNM. TSIDPLB writes the label from the same area.

To Read a Label

Step	Action
1	Move the DTF name of the label to be processed to FILENAME.
2	Call TSIDGLB, specifying FILENAME and LBLRETCB.

TSIDGLB Example

```
...
MVC FILENAME,=CL7'INPFILE'
CALL TSIDGLB,(FILENAME,LBLRETCB)
...
COPY TSIDGLB
...
```

To Write a Label

Step	Action
1	Read the label using the TSIDGLB procedure.
2	Modify the label information (see warning below).
3	Move the DTF name of the label to be processed to FILENAME.
4	Move the length of the label to LBLLEN (80 for tape; 200 for disk).
5	Call TSIDPLB, specifying FILENAME and LBLRETCB.

WARNING!

TSIDPLB does not validate the data on the label you write, so you must ensure that it is correct. Although the label can be written successfully, results are unpredictable if its data is incorrect.

TSIDPLB Example

```
...
MVC FILENAME,=CL7'OUTFILE'
MVC LBLLEN,=H'80'
CALL TSIDPLB,(FILENAME,LBLRETCB)
...
COPY TSIDGLB
...
```

9

Job Management Control Language

BIM-EPIC's Job Management Control Language (JMCL) gives you greater control over the execution of your job. The JMCL facility is activated using the JCLLANG=YES configuration option as described in the *BIM-EPIC Installation and System Guide*.

Chapter Contents

JMCL Statements	9-2
JMCL Parameters.....	9-5
Using the JMCL Interface (TSIDJC2)	9-15
Executing CP Commands.....	9-20

JMCL Statements

Introduction

Description

JMCL parameters are specified on VSE JCL statements. You can specify JMCL parameters on:

- EXEC statements
- JOB statements
- OPTION statements
- Comment statements

Most JMCL parameters are valid on only one or two of the JCL statement types. Check the description of the particular parameter to see where it can be specified.

JMCL parameters are in keyword format. You can specify more than one JMCL parameter on a single statement. Parameters must be separated with commas. Because many JMCL parameters are interdependent, be sure to check the end of this section for an example of the parameters working together.

Using JOB, EXEC, and OPTION Statements

Description

On JOB, EXEC, and OPTION statements, the JMCL parameters must:

- follow at least one VSE keyword (or a comma between two spaces)
 - precede any comments
-

Syntax

```
// JOB jobname parameters comments  
// EXEC program parameters comments  
// OPTION option parameters comments
```

Example

In this example, the job name is PAYROLL. RESET is a JMCL parameter in its correct position. CHECKS is a comment.

```
// JOB PAYROLL RESET CHECKS
```

Using Comment Statements

Description

VSE comment statements can invoke JMCL by using special JMCL parameters.

Syntax

* JMCL parameters

Example

* JMCL NAME=BGSTEP

JMCL Parameters

JCL Syntax Checking

Description

With JMCL, you can retain datasets that have been flagged for purge (HOLD). You can cancel a job if it exceeds a specified time limit (TIME=n) or specify special processing for restarting canceled jobs (TYPRUN=RESTART).

In addition, you can prevent some job cancellations by requesting DD statement syntax checking (TYPRUN=SCAN and TYPRUN=SCANS).

HOLD

Use HOLD to prevent datasets that have been flagged for purge at EOJ from being purged if the job cancels.

HOLD is valid on JMCL statements and is effective for the duration of the job.

Syntax

HOLD

TIME

Use TIME to specify a time limit for job execution. If your job exceeds the limit you specify, it is terminated without a dump and the "CC" condition code (see "Conditions and IF") is set to 1.

TIME is valid on JOB statements.

WARNING!

The TIME parameter will be overridden by any program which uses IBM's SETIME instruction. If this occurs, your job may not cancel as expected.

Syntax

TIME=nnnn

nnnn specifies the time period in minutes. Valid time periods are from 1 minute to 1440 minutes. No error message is issued if you specify a number out of this range. If you specify a number greater than 1440, the time limit is set to 1440 (unlimited time execution).

Example

```
// JOB XYZ TIME=60
```

TYPRUN

TYPRUN has two separate uses. It can be used to specify:

- syntax checking for DD statements
- special processing for job restarts

TYPRUN is valid on JOB statements.

Syntax
$$\text{TYPRUN} = \left\{ \begin{array}{l} \text{SCAN} \\ \text{SCANS} \\ \text{RESTART} \end{array} \right\}$$

**TYPRUN
Parameters**

SCAN checks DD statement syntax and prints all JCL and any error messages. SCAN scans the JCL until the first /& statement, after which normal processing resumes. The job is not executed.

If there is SYSIPT data in your job stream you must also specify the DATA parameter. See 9-8 in this manual for more information.

SCANS checks DD statement syntax and prints only DD statements in error and the resulting error messages. The job is not executed. SCANS scans the JCL until the first /& statement, after which normal processing resumes.

If there is SYSIPT data in your job stream you must also specify the DATA parameter. See 9-8 in this manual for more information.

RESTART When a job cancels, you may want to restart it at a job step other than the first step of the job. You can branch to the desired step with a JMCL GOTO parameter. However, if you have specified the JMCL parameter HOLD or the configuration option PURBYP=YES, disk datasets flagged for purge are not purged when a job cancels. If you bypass steps with GOTO, work datasets created in those steps need to be specially purged.

TYPRUN=RESTART purges those datasets when they are flagged for purge with the DLBL option P or X or DISP=(x,DELETE).

Example

```
// JOB XYZ TYPRUN=SCANS
```


Condition Checking and Branching

Description

The GOTO facility allows you to bypass JCL statements without deleting them or changing them to comments before you submit the job. Condition checking, with the IF parameter, allows you to check whether certain conditions exist before you branch to a new part of the job.

GOTO

GOTO supports unconditional branching and conditional branching when used in conjunction with the JMCL IF parameter. Use GOTO for unconditional branching when you are restarting a job.

GOTO only branches forward in a job. GOTO cannot branch back to an earlier statement.

Mark the destination of the branch with the JMCL NAME parameter. If you bypass SYSIPT data during the branch, you must also use the DATA parameter. See "NAME" and "DATA" below.

If you are restarting a job, you may need to use the TYPRUN=RESTART parameter. Please note that if the step you execute in the restart checks for a condition code, that code may be invalid at restart time.

GOTO is valid on OPTION and JMCL statements.

Syntax

`GOTO=label`

label specifies the NAME label to branch to. The value must match the label specified on the NAME parameter (see below). If you specify GOTO=JOBEND, GOTO branches to end-of-job (/&) and a NAME parameter is not required.

Example

```
* JMCL GOTO=SORT3
```

NAME

Use NAME to mark the destination of a GOTO branch. Place the NAME statement immediately before the statement at which you want to resume processing.

NAME is valid on JMCL statements and EXEC statements.

Syntax

`NAME=label`

label specifies a label name. The value specified here must match the label specified on the GOTO parameter.

Example

```
* JMCL NAME=SORT3
```

DATA

Use DATA to bypass SYSIPT data during a GOTO branch. Because SYSIPT data is processed specially, BIM-EPIC cannot bypass it automatically.

Place a DATA parameter on the EXEC statement of each program that is followed by SYSIPT data. DATA is valid only on EXEC statements.

Syntax

DATA [=*n*]

**Branching
Subparameters**

Parameter	Required?	Valid	Default
GOTO	No	1 to 8 characters	None
NAME	No	1 to 8 characters	None
DATA	No	1 to 127	1

n specifies the number of SYSIPT data streams to bypass for a single EXEC. The default is one, so you only need to specify *n* when there is more than one data stream for the EXEC.

Example

```
// EXEC PAYROLL DATA
<SYSIPT data here>
/*
```

Conditions and IF

Description

Use IF to test for a condition. IF can test the current condition code, the highest condition code, the VSE cancel code, the date, time, CPU, partition, and a global constant you set. You can also set the current and highest condition codes. See 9-11 in this manual for more information.

You can test multiple conditions using the IF parameter. Both "AND" and "OR" conditions are supported.

When IF is used with GOTO, the branch occurs if the condition is true. When IF is used on an EXEC, the step is executed only if the condition is true. If it is false, BIM-EPIC bypasses the EXEC.

The IF parameter has three subparameters, which must be separated by at least one space. To test AND conditions, specify separate sets of subparameters on the same IF parameter and separate them with commas. To test OR conditions, use separate IF parameters.

When used with the GOTO parameter, IF is valid on OPTION and JMCL statements. It is also valid on EXEC statements, without GOTO. "AND" conditions are valid on OPTION, EXEC, and JMCL statements. "OR" conditions are valid on OPTION and JMCL statements only.

Syntax

```
IF=(type operand value [ ,type operand value, ... ] )
```

Parameters

For...	Valid values are...
CC, HC, and DOSCC	1 to 999
PART	2 character partition ID
CPU	0 to 9, A to Z
DATE and TIME	2 to 8 characters
Gnnn	1 to 3 characters

type specifies the type of input data to be tested.

Use...	To test the...
CC	current condition code
CPU	CPU ID
HC	highest condition code
DOSCC	VSE cancel code
PART	partition ID

DATE	current date
TIME	current time
Gnn	global constant

operand specifies the kind of test.

Use...	To test whether the input data is...
EQ	equal to <i>value</i>
NE	not equal to <i>value</i>
GT	greater than <i>value</i>
LT	less than <i>value</i>
GE	greater than or equal to <i>value</i>
LE	less than or equal to <i>value</i>

value specifies the constant to be used in the comparison.

Global constants are set with the SETG JMCL parameter. Identify the global constant you are testing by specifying 01 through 15 for nn.

DOSCC values are decimal codes generated by VSE. For normal EOJ the code is 16. You can check the DOSCC across job boundaries.

The CPU ID is the same number generated when you use CPU independence—that is, the last digit specified for the CPUIDS configuration option.

In partition comparisons, only the EQ and NE operands are valid. For example IF=(PART EQ BG) is valid. IF=(PART LE BG) is invalid.

Date is specified in yy-mm-dd format. Time is specified in hh-mm-ss format. If you wish, you can test only a portion of the date and time. To do so, supply only the portion you wish to test. Supply asterisks (*) as filler for beginning parts of the field. For example, to test for the first day of the month, code IF=(DATE EQ **-*-01). To test for January, code IF=(DATE EQ **-01).

Example

```
* JMCL IF=(HC EQ 99),GOTO=JOBEND
* JMCL IF=(PART NE BG,CPU EQ 5),GOTO=BGSTEP
// EXEC PHASE1 IF=(CC LT 16)
```

The first statement tests the highest condition code. If the highest condition code is 99, BIM-EPIC branches to the end of the job. The second statement uses an "AND" condition. If the job is not running in the BG partition but is running on CPU 5, a branch is made to BGSTEP. If neither of these tests is true, the job continues with the EXEC statement. However, PHASE1 only executes if the current condition code is less than 16.

Setting Values

Description

You can establish a number of constants with JMCL parameters. These constants can be tested with the IF parameter.

Use SETC to set the current condition code. Use SETG to set up to 15 global constants.

SETC

SETC sets the current condition code to the number you specify. This also updates the highest condition code, if appropriate.

The current condition code remains in effect until a new value is set by another SETC parameter, until a RESET is encountered, or until a new value is set by the TSIDJC2 subroutine.

SETC is valid on OPTION and JMCL statements, where it is in effect as soon as the statement is processed. It is also valid on EXEC statements, where the condition code is set only if the program executes successfully.

Syntax

`SETC=nnn`

nnn specifies the number for the current condition code.

SETG

SETG sets global constants. You can have up to 15 global constants. Each constant remains in effect until you IPL VSE or until you set a new value with SETG.

SETG is valid on JMCL and OPTION statements only.

Syntax

`SETGnn= 'constant'`

nn identifies which global constant you are setting. Valid constants are 1-15 or 1-16 characters.

constant specifies the value of the constant. You can use spaces.

Parameters

Parameter	Required?	Valid	Default
<i>nnn</i>	No	1 to 255	None
<i>nn</i>	No	01 to 15	None
<i>constant</i>		1 to 16 characters	None

Symbolic Replacements

Description

Symbolic replacements "replace" a set of characters that are found in your JCL.

Symbolic replacements are useful for JCL subparameters or comments.

BIM-EPIC inserts the replacement value into the statement and shifts to the right any data following the replacement symbol. For example:

```
* JMCL &1='INVENTORY.INPUT'
// TLBL SORTOUT,'&1',,,,,,2
```

would result in:

```
// TLBL SORTOUT,'INVENTORY.INPUT',,,,,,2
```

WARNING!

You must take care to avoid truncation of the statement. The statement must not exceed 71 columns with the replacement value inserted.

Symbolic replacements are not valid on POWER JECL statements or in SYSIPT data. Any other statement (including JMCL statements) can use a symbolic replacements.

Assign values to symbolic replacements on OPTION statements.

&n

The symbolic replacement parameter has the format "&n", where *n* is a number between 0 and 9. Use it to replace characters with the value you assign to the symbol.

You can have up to 10 of these parameters in effect at the same time. Each one remains in effect until it is re-assigned in a JMCL or OPTION statement or it is nullified with RESET, or end-of-job occurs.

Syntax

```
&n='constant'
```

n identifies symbol being assigned.

constant specifies the value assigned to the symbol. The symbol must be enclosed in apostrophes and can contain spaces.

Example

In the following example, the DLBL is changed to

```
// DLBL INPUT,'PAYROLL.INPUT'
```

```
* JMCL &1='PAYROLL.INPUT'
// DLBL INPUT,'&1'
```

RESET

Description

RESET nullifies most JMCL parameters. You can use it to set the current or highest condition code to zero, nullify GOTO branches, and nullify TIME control over a job.

We recommend that you use the global RESET parameter on the JOB statements and at the end of all jobs that do not have a logical dependency on previous jobs.

RESET is valid on JOB, EXEC, JMCL, and OPTION statements. When it is specified on a JOB statement, it is processed before any other JMCL parameters.

Syntax

$$\text{RESET} = \begin{bmatrix} \text{CC} \\ \text{HC} \\ \text{NAME} \\ \text{TIME} \end{bmatrix}$$

Subparameters

CC resets the current condition code only.

HC resets the highest condition code only.

NAME nullifies any branch in effect. Processing resumes with the current statement.

TIME nullifies TIME control over the job.

When no subparameters are specified, all values are reset to the defaults.

JMCL Example

Description

The following example shows JMCL parameters working together.

```
// JOB EXAMPLE TYPRUN=RESTART,RESET,TIME=240
* JMCL HOLD
// OPTION LOG &1=PAYROLL
* JMCL &2=PAYROLL
* JMCL GOTO=STEP2
// ASSGN SYS001,DISK,VOL=VOLUME,SHR
// DLBL SORTWK1,'=.SORTWK1(P)'
// EXTENT SYS001
// DLBL SORTIN,'&1.INPUT'
// DLBL SORTOUT,'&2.OUTPUT'
// EXEC SORT DATA
sort statements
/*
* JMCL NAME=STEP2
* JMCL IF=(DATE EQ ***--01,TIME LT 19-00-00),GOTO=STEP3
// DLBL INPUT,'&1.INPUT'
statements
* JMCL NAME=STEP3
statements
/&
```

This example is a restart job.

The RESET parameter is included to reset any JMCL parameters established in a previous job. The TIME parameter limits the execution of this job to 4 hours.

Since it is a restart job, the GOTO bypasses the first step, which completed successfully in a previous execution. (GOTO is placed after the symbolic replacement definitions, so that they remain in effect.) The DATA parameter allows BIM-EPIC to bypass the sort control statements (SYSIPT data). The TYPRUN=RESTART parameter is specified so that the sort work area in the bypassed step is deleted. It was retained in the previous execution because of the JMCL HOLD parameter, specified in the second statement.

The * JMCL NAME=STEP2 statement defines the point at which processing should resume after the GOTO branch. The next statement tests the date and time. If it is the first of the month and earlier than 7:00 p.m., BIM-EPIC branches to the statement * JMCL NAME=STEP3. If either condition is false, processing continues with the // DLBL INPUT statement.

Both the &1 and &2 symbols are replaced with PAYROLL.

Using the JMCL Interface (TSIDJC2)

Introduction

TSIDJC2

A batch program can interface with BIM-EPIC's Job Management Control Language by calling the TSIDJC2 subroutine. TSIDJC2 can be used to retrieve the following information:

- DOS job name
- program name
- current date (MM/DD/YY)
- current time (HH.MM.SS)
- cpu id character
- current JMCL condition code
- highest JMCL condition code
- task time limit
- JMCL symbolic values
- JMCL global constants

TSIDJC2 can update the following information:

- current JMCL condition code
 - highest JMCL condition code
 - JMCL symbolic values
 - JMCL global constants
-

Parameter List

TSIDJC2 uses a parameter list to indicate the function being performed (Get or Put) in addition to providing the data areas for returning or updating data fields. Two modules are provided in the source library:

- TSIDJC2.C for COBOL programs
- TSIDJC2.A for Assembler programs

The TSIDJC2 subroutine can also be called from any language that uses standard calling conventions.

Retrieving JMCL Data (GET REQUEST)

GET REQUEST (G) Data can be retrieved from the JMCL system by passing a character 'G' to TSIDJC2 in the request byte of the parameter list (field JCSREQ in TSIDJC2). And JCS-REQ in TSIDJC2.C Control is returned to the next instruction following the CALL.

COBOL Example

```
WORKING-STORAGE SECTION.  
COPY TSIDJC2.  
.  
.  
.  
PROCEDURE DIVISION.  
MOVE 'G' TO JCS-REQ.  
CALL 'TSIDJC2' USING JCS-START.
```

Assembler Example

```
MVI JCSREQ,JCSREQG  
CALL TSIDJC2,(JCSSTART)
```

Updating JMCL Data (PUT REQUEST)

PUT REQUEST (P)

JMCL data can be updated by passing a character 'P' to TSIDJC2 in the request byte of the parameter. This byte is labeled JCSREQ in TSIDJC2.A and JCS-REQ in TSIDJC2.C. A Put request updates the current JMCL condition code, the highest JMCL condition code, all JMCL symbolics and all global constants. All Put requests should be preceded by a Get request to ensure that all data fields that are not explicitly being modified are updated to their previous data values. Control is returned to the next instruction following the CALL.

To update current or highest JMCL condition code

Move the new condition code to the appropriate data field as a 2-byte binary value. During the update process, the highest condition code is set first. Then the current condition code is set. When the current condition code is set, it is checked against the highest condition code to determine whether the new current condition code is now also the new highest condition code. The JMCL current condition code is labeled JCSCC in TSIDJC2.A and JCS-CURR-COND-CODE in TSIDJC2.C. The JMCL highest condition code is labeled JCSCC in TSIDJC2.A and JCS-HIGH-COND-CODE in TSIDJC2.C.

To update a symbolic

Move the new 1 - 20 character symbolic into its appropriate data field. Symbolics &0 thru &9 are labeled JCSSYM0 thru JCSSYM9 in A.TSIDJC2 and JCS-SYMBOLIC-0 thru JCS-SYMBOLIC-9 in C.TSIDJC2. If the new symbolic is less than 20 characters, it must be left-justified and padded with spaces on the right. To remove a symbolic, clear the symbolic data field with hex zeros or LOW-VALUES.

To update a global constant

Move the new 1 - 16 character constant into its appropriate data field. Global constants G01 thru G15 correspond to the parameter data fields JCSCGBL1 thru JCSCGBL15 in A.TSIDJC2 and JCS-GLOBAL-CONS-1 thru JCS-GLOBAL-CONS-15 in C.TSIDJC2. If the new global constant is less than 16 characters, it must be left justified and padded with spaces on the right. To remove a global constant, clear the global constant data field with hex zeros or LOW-VALUES.

COBOL Example

```
WORKING-STORAGE SECTION.
COPY TSIDJC2.
.
.
.
PROCEDURE DIVISION.
MOVE 'G' TO JCS-REQ.
CALL 'TSIDJC2' USING JCS-START.
MOVE 0 TO JCS-CURR-COND-CODE.
MOVE 'MONTHLY' TO JCS-SYMBOLIC-2.
MOVE LOW-VALUES TO JCS-GLOBAL-CONSTANT-10.
MOVE 'P' TO JCS-REQ.
CALL 'TSIDJC2' USING JCS-START.
```

**Assembler
Example**

```
MVI    JCSREQ,JCSREQG
CALL   TSIDJC2,(JCSSTART)
MVC    JCSCC,=XL2'00'
MVC    JCSSYM2,=CL20'MONTHLY'
XC      JC SGBL10,JCSGBL10
MVI    JCSREQ,JCSREQP
CALL   TSIDJC2,(JCSSTART)
```

TSIDJC2 Return Codes

Description

TSIDJC2 returns a value in the return code byte of the parameter list. This byte is labeled JCSRC in TSIDJC2.A and JCS-RC in TSIDJC2.C. All return codes are one-byte character values.

Values

Code	Explanation
0	Successful completion of the G(et) or P(ut) request.
1	Either BIM-EPIC or the JMCL system are not active.
2	A secondary routine (TSIDJCS) could not be loaded from the BIM-EPIC system library. The normal cause of this problem is not having the BIM-EPIC system library in the search chain of the partition.
3	There was insufficient Partition GETVIS available to load the TSIDJCS routine. The simplest way to correct this problem is to add a SIZE= parameter to the EXEC statement.
4	The request code in JCSREQ or JCS-REQ was not a G or P.

Executing CP Commands

Description

You can execute any VM/CP command in your VSE job stream by using JMCL's CP control statement.

The CP control statement must begin in column 1.

Configuration Options

For CP commands to be valid, the VM configuration option must be set to YES.

Syntax

* CP *command*

where *command* specifies which VM/CP command to execute.

Example

The following statement lists all tape drives on the operator console.

```
* CP Q T ALL
```

10

Job Scheduling

BIM-EPIC's job scheduling system can be used to control execution across all partitions and VSE machines sharing the same DSN Catalog.

Chapter Contents

Event Posting	10-2
Introduction	10-2
Scheduling Functions	10-3
Posting an Event and Releasing Matching Jobs (TSIDREL and TSIDCREL)	10-4
Posting an Event Only (TSIDLOC)	10-5
Checking for a Posted Event (TSIDCHK and TSIDCCHK)	10-6
Deleting a Posted Event (TSIDUNL)	10-7
Executing in Batch Jobs.....	10-8
Executing Standard Processing Subroutines	10-9
Processing Requirements for TSIDREL, TSIDLOC, and TSIDUNL.....	10-9
Processing Requirements for TSIDCHK	10-10
Executing in CICS.....	10-11
Using the Subroutines	10-11
Return Codes	10-12
Processing Requirements	10-14
Creating Job Setup/Run Sheets (TSIDFAV)	10-15

Event Posting

Introduction

Description

When a job, step, or process has completed, you can direct BIM-EPIC to **post an event**. BIM-EPIC writes the name of the event in the DSN Catalog as a version of the dataset *EPIC.EVENT.DSN*. You assign an event a name, 1 to 8 characters long. You then use this name to identify which event has completed.

Like versions of a regular dataset, posted events become eligible for scratch according to the retention criteria cataloged for the dataset.

If *EPIC.EVENT.DSN* is not already in the catalog, BIM-EPIC adds it automatically the first time an event is posted. Then automatic day and cycle retention is used to retain posted events. You may want to change this retention with manual catalog maintenance.

Posted events can be deleted with the SCRATCH functions of TSIDUTL or TSIDMNT. They are normally deleted using the TSIDUTL utility.

Job scheduling subroutines put in the event name in the SYSPARM value, and COBOL programs do special processing when SYSPARM values include A or D. Avert potential conflict by not using these letters in event names processed with job scheduling subroutines.

Scheduling Functions

Description

There are four job scheduling functions:

- Posting an event and releasing all matching jobs
- Posting an event only
- Checking for a posted event and releasing the requesting job
- Deleting a posted event

For each job scheduling function, BIM-EPIC provides one batch utility, one standard processing subroutine, and one CICS subroutine.

The batch utilities and subroutines share the same names. CICS subroutine names begin with TSIDC.

Posting an Event and Releasing Matching Jobs (TSIDREL and TSIDCREL)

Description

TSIDREL and TSIDCREL post events and automatically release jobs with matching job names from the POWER reader queue.

To be released, a job must be in disposition "H" or "L" in the POWER reader queue, and its job name must match the event name posted.

If the event name is 8 characters, BIM-EPIC releases the job with the POWER command:

R RDR,*event-name*

If the event name is less than 8 characters, it releases the job with:

R RDR,**event-name*

If jobs are released into the same partition and have the same priority, they execute in the order they are released.

Posting an Event Only (TSIDLOC)

Description

TSIDLOC and TSIDCLOC post events. They do ***not*** release jobs. When you use TSIDLOC or TSIDCLOC, dependent jobs must themselves issue a check for the posted event using TSIDCHK or TSIDCCHK.

Checking for a Posted Event (TSIDCHK and TSIDCCHK)

Description

TSIDCHK and TSIDCCHK are used within a job to check to see if an event has been posted before continuing with a dependent task. If the event is posted your job continues uninterrupted.

If you check for the event with TSIDCHK (in either batch or subroutine) and the event is *not* posted, execution is interrupted. TSIDCHK issues a message to the operator and ask whether the job should be canceled or suspended. If the operator responds WAIT, the job is suspended, and the partition is stopped until the event is posted. When the event is posted, the partition is restarted automatically.

The CICS version, TSIDCCHK, does *not* interrupt processing if the event is not posted. You must check the return code passed, and process accordingly.

Deleting a Posted Event (TSIDUNL)

Description

TSIDUNL and TSIDCUNL delete posted events. Use them to delete an event after all dependent tasks have completed. This is important if another processing cycle occurs before the event is scratched automatically.

Remember that if you do not delete them, events become eligible for scratch as described under "About Posted Events".

Executing in Batch Jobs

Description

TSIDREL, TSIDLOC, TSIDCHK, and TSIDUNL can be executed in batch jobs.

JCL Requirements

```
// OPTION SYSPARM='event-name '  
// EXEC TSIDxxx  
// OPTION SYSPARM='          '
```

Control Statements

Two OPTION statements are required: one to specify the event name as a SYSPARM and one to clear the SYSPARM so it does not affect later processing.

event-name specifies the name of the event to be posted (the name of the job to be released). It may be 1 to 8 characters long.

TSIDxxx specifies the utility you are executing. Specify TSIDREL, TSIDLOC, TSIDCHK, or TSIDUNL.

Example

This job executes TSIDREL. It posts an event with the name PAY and releases all jobs in the POWER reader queue which have job names beginning with PAY.

```
// JOB RELEASE JOBS  
// OPTION SYSPARM='PAY'  
// EXEC TSIDREL  
// OPTION SYSPARM='          '  
/&
```

Executing Standard Processing Subroutines

TSIDREL, TSIDLOC, TSIDCHK, and TSIDUNL are callable subroutines for non-CICS programs. Standard linkage conventions apply to all four subroutines.

Processing Requirements for TSIDREL, TSIDLOC, and TSIDUNL

Procedure

Step	Action
1	Perform the “event” task.
2	Move the event name to the EVENT-NAME field. See "Posted Events," for details about event name limitations.
3	Issue a call to the appropriate subroutine.

Example

This task calls TSIDUNL. It deletes the PAY event after PAYROLL checks have been processed.

```
...  
WORKING-STORAGE SECTION.  
01 EVENT-NAME          PIC X(8) VALUE SPACES.  
PROCEDURE DIVISION.  
    PERFORM PAYROLL-CHECKS.  
    MOVE 'PAY' TO EVENT-NAME.  
    CALL 'TSIDUNL' USING EVENT-NAME.  
...
```

Processing Requirements for TSIDCHK

Procedure

Step	Action
1	Move the event name to the EVENT-NAME field. See "Posted Events," for details about event-name limitations.
2	Issue a call to TSIDCHK.
3	Perform the dependent task.

Example

The following task calls TSIDCHK, with PAY as the EVENT-NAME. If that event is posted, the task performs PAYROLL-CHECKS.

```
...  
WORKING-STORAGE SECTION.  
01 EVENT-NAME          PIC X(8) VALUE SPACES.  
PROCEDURE DIVISION.  
    MOVE 'PAY' TO EVENT-NAME.  
    CALL 'TSIDCHK' USING EVENT-NAME.  
...
```


Executing in CICS

Using the Subroutines

Description

TSIDCREL, TSIDCLOC, TSIDCCHK, and TSIDCUNL are CICS subroutines that can be called from a CICS command-level program.

The program must link to them using a 69-byte communications area with the following format:

Byte	Contents
1–8	EVENT-NAME
9	EVENT-RETURN-CODE
10–69	EVENT-POWER-MESSAGE

For TSIDCREL, when EVENT-RETURN-CODE is "P", bytes 10 through 69 contain the following:

Byte	Contents
10–12	CTL Spool Return Code
13–21	reserved
15–22	XPCC Function That Failed
23–36	reserved
37–38	VSE Return Code
39	reserved
40–41	Reason Code
42–54	reserved
55–56	POWER Return Code
57	reserved
58–59	POWER Feedback Code
60–69	reserved

Return Codes

TSIDCREL

Code	Explanation
A	The event has already been posted. No jobs are released.
F	The DSN Catalog is full and the event cannot be posted.
M	The event name specified in EVENT-NAME is missing or invalid.
P	The event was posted, but there was an error in releasing jobs from the POWER reader queue.
W	The event was posted, but POWER is in a busy state and jobs were not released.
X	The event was posted, but the XECB supervisor table is full.
Y	The event was posted and the jobs were released successfully.
5	The event was posted, but the POWER XPCC function failed.
6	The event was posted, but POWER is being terminated.
7	The event was posted, but POWER has terminated abnormally.
8	The event was posted, but TSIDCREL could not make a connection to POWER within 30 seconds.
9	The event was posted, but the POWER file could not be located.

Return codes 5–9 indicate highly unusual situations and should not occur.

Each of the numeric return codes (*n*) is associated with a BIM-EPIC message, numbered EPV3*n*. For example, to find the message associated with return code 5, see message EPV35 in *Messages and Codes*.

TSIDCLOC

Code	Explanation
A	The event has already been posted.
F	The DSN Catalog is full and the event cannot be posted.
M	The event name is missing or invalid.
Y	The event was posted successfully.

TSIDCCHK

Code	Explanation
M	The event name is missing or invalid.
N	The event is not posted.
Y	The event is posted.

TSIDCUNL

Code	Explanation
E	The *EPIC.EVENT.DSN* dataset definition has been deleted from the DSN Catalog.
F	The DSN Catalog is full.
M	The event name is missing or invalid.
N	The event was not posted.
Y	The event was deleted successfully.

Processing Requirements

Procedure

Step	Action
1	Move the event name to the EVENT-NAME field. See "Posted Events," for details about event-name limitations.
2	Link to the subroutine.
3	Check the return code passed and process accordingly.

Example

```
WORKING-STORAGE SECTION.  
01 EVENT.  
    02 EVENT-NAME          PIC X(08).  
    02 EVENT-RETURN-CODE   PIC X(01).  
        88 SUCCESS        VALUE 'Y'.  
        88 FAILURE        VALUE 'N'.  
    02 EVENT-POWER-MESSAGE PIC X(60)  
PROCEDURE DIVISION.  
    MOVE 'PAY' TO EVENT-NAME.  
    EXEC CICS  
        LINK PROGRAM ('TSIDCCHK')  
            COMMAREA (EVENT)  
            LENGTH (69)  
    END EXEC.  
    IF SUCCESS  
        GOTO PERFORM  
    ELSE  
        GOTO WAIT.  
    ...
```

This example checks for a posted event with TSIDCCHK. If the event is posted, it performs the dependent task. If the event is not posted, it goes to the WAIT routine.

Creating Job Setup/Run Sheets (TSIDFAV)

UPSI Settings

```
// UPSI  1XXXX Print comment cards.
// UPSI  X1XXX Print user data cards.
// UPSI  XX1XX Scan all VSE jobs for creation of input files.
// UPSI  XXX1X Scan all POWER JOBS for creation of input files.
// UPSI  XXXX1 Input JCL submitted backwards (POWER MODE).
```

Modes of Execution

TSIDFAV can be executed in VSE mode or in POWER mode. The execution mode is determined by UPSI bit 4 (relative to 0).

To execute in VSE mode, set this UPSI bit to 0 (off).

To execute in POWER mode, set this UPSI bit to 1 (on).

SET Control Statement

The SET control statement can be used to define the partition ID and CPUID values used to replace equal signs (=) in the dataset name. The SET control statement must precede all job controls submitted. It must begin on column 1 whether execution is in VSE or POWER mode.

Format

```
SET PART=xx CPUID=y
```

Parameter Values Explained

xx defines the 2-character partition ID for replacing the first two equal signs. If this parameter is omitted, the first two equal signs are replaced by the partition ID where TSIDFAV is being executed.

y defines the 1-digit CPUID for replacing the third equal sign. If this parameter is omitted, the third equal sign is replaced by the last digit of the CPUID where TSIDFAV is being executed.

VSE Mode

In VSE mode, all job control is submitted to TSIDFAV in the order in which it is executed. The JCL is submitted either as instream data (following the EXEC statement), or via tape or disk input using the DTF name DISKIN. If POWER JECL is included in the jobstream, the input must be submitted via tape or disk. If the input is submitted after the EXEC statement, SYS004 must be assigned to SYSIPT. Under all types of input, the last record must be the EOF flag as determined by TYPE=EOF in the TSIDFAV macro, or defaulted to two dashes (- -).

**JCL Examples for
VSE Mode**

```
// JOB TSIDFAV
* INPUT INSTREAM DATA
// ASSGN SYS004,SYSIPT          REQUIRED
// UPSI XXXX0
// EXEC TSIDFAV
  VSE JCL
--
/*
/&

// JOB TSIDFAV
* DISK INPUT
// DLBL DISKIN,'FAV.INPUT.JCL'   REQUIRED
// ASSGN SYS004,DISK,...
// UPSI XXXX0
// EXEC TSIDFAV
/*
```

POWER Mode

In POWER mode, the job control must be submitted backwards (in reverse order of execution) and upside down so that card column 1 is submitted to TSIDFAV in card column 80. This prevents POWER from processing any POWER JECL submitted as instream data. The JCL can be submitted either as instream data (following the EXEC statement), or via tape or disk input using the DTF name DISKIN. If the input is submitted after the EXEC statement, SYS004 must be assigned to SYSIPT.

The EOF flag must be submitted in columns 1 and 2 (not reversed).

**JCL Examples for
POWER Mode**

```
// JOB TSIDFAV
* INPUT INSTREAM DATA
// ASSGN SYS004,SYSIPT          REQUIRED
// UPSI XXXX1
// EXEC TSIDFAV
  VSE JCL AND POWER JECL READ WITH 9 EDGE IN
--                                (EOF FLAG)
/*
/&

// JOB TSIDFAV
* DISK INPUT
// DLBL DISKIN,'FAV.INPUT.JCL'   REQUIRED
// ASSGN SYS004,DISK,...
// UPSI XXXX1
// EXEC TSIDFAV
/*
```

TSIDFAV reads the VSE procedures and POWER SLIs from the appropriate VSE library. However, TSIDFAV does not support either PROC or SLI overwrites. Nested SLIs are supported, but nested PROCs are not supported.

11

Job Activity Reports

BIM-EPIC provides several job accounting programs. These programs produce job accounting statistics using data collected from the Recorder File. All job accounting is affected by the JOBACCT and RECORD configuration options.

Chapter Contents

Recorder History Dataset.....	11-2
Creating the Activity Log Report (TSIDARP)	11-3
Creating the DSN/Job Cross-Reference Report (TSIDBRP).....	11-6
Job Accounting Reports by Partition (TSIDTAC).....	11-9

Recorder History Dataset

Description

Recorder File backups can be used to create job activity reports. For example, you may want to combine daily backups into a single Recorder History dataset. You can create a new Recorder History dataset weekly, monthly, or for whatever time period you wish.

Step 1

Create a new Recorder History dataset, using a tape-to-tape copying utility such as TSIDDDTD. The Recorder backup format is fixed-blocked, with a record length of 338 and a block size of 5746.

```
// JOB INITIALIZE RECORDER HISTORY FILE
// TLBL INPUT, 'DAILY.RECORDER'
// TLBL OUTPUT, 'RECORDER.HISTORY'
// EXEC TSIDDDTD
/*
/ &
```

Step 2

Combine the new Recorder File backups into an existing Recorder History dataset. The example below uses a sort program, but you can use whatever mechanism you prefer.

```
// JOB MERGE VSE RECORDER FILES
// TLBL SORTIN1, 'VSE.DAILY.BACKUP',,,,,,2
// TLBL SORTIN2, 'VSE.MNTHLY.BACKUP',,,,,,2
// TLBL SORTOUT, 'VSE.MNTHLY.BACKUP',,,,,,2
// DLBL SORTWK1, '===.SORTWK1'
// EXTENT SYS004
// EXEC TSIDSRT
  SORT FIELDS=(25,2,A,21,6,A,17,4,A),FILES=2,FORMAT=BI
  RECORD TYPE=F,LENGTH=338
  INPFIL BLKSIZE=5746
  OUTFIL BLKSIZE=5746
  OPTION PRINT=ALL
  END
/*
/ &
```


Creating the Activity Log Report (TSIDARP)

Description

TSIDARP produces the Activity Log report, which shows open and close activity for controlled datasets. Open and close activity is sorted by:

- System ID
- Date and time
- Job name

The Activity Log report can be limited to activity generated on any CPU or occurring during a specific time period. TSIDARP also produces the cross-reference dataset that is used as input to TSIDBRP.

Control Statements

There are two control statements for TSIDARP: EXCLUDE and TSIDARP. Both are optional.

JCL Requirements

```
// ASSGN SYS003,SYSIPT
// ASSGN SYS004,SYSLST
// ASSGN SYS001,DISK,VOL=volser,SHR
// TLBL SYS005,'daily.recorder.backup'
// DLBL SYS002,'recorder.cross.ref'
// DLBL SORTWK1,'===.sortwk1'
// EXTENT SYS001
// EXEC TSIDARP
control statements
/*
```

UPSI Values

None

Dataset Requirements

SYS005 identifies the dataset to be used as input. You can use the Recorder backup dataset created by TSIDRFS, or you can use a Recorder history dataset.

SYS002 identifies the cross-reference dataset produced as output. This dataset is used as input for TSIDBRP.

SORTWK1 identifies a sort work area for an internal sort. You must provide an ASSGN and a dummy EXTENT statement.

EXCLUDE Control Statement

Description	<hr/> <p>The EXCLUDE control statement specifies a job to be excluded from the report. If used, it must come before the TSIDARP statement. You can use up to 20 EXCLUDE statements, each statement specifying one 1 to 8–character job name.</p> <hr/>
Syntax	<p>EXCLUDE= <i>jobname</i></p>

TSIDARP Control Statement

Description

The TSIDARP control statement specifies what dates, CPUs, and partitions to include in the report. It is optional, and if it is omitted, the report contains all relevant data.

The TSIDARP statement is a 29-byte record, consisting of five fields:

Bytes	Contents
1–7	'TSIDARP'
8–14	Starting date, in Julian format (ccyyddd)
15–21	Ending date, in Julian format (ccyyddd)
22–27	CPU ID
28–29	Partition ID

The statement must begin with 'TSIDARP'. All other fields are optional. To omit a field, leave the field blank. If a field is omitted, the report will contain all relevant data for that field.

Syntax

TSIDARPssssssseeeeeeiippi

Example

The following job produces an Activity Log report for jobs run from December 1, 1994 to December 31, 1994. It includes only those jobs that were run on CPU 55555 in the BG partition. It excludes all datasets from the CICSPR16 job.

```
// JOB TSIDARP ACTIVITY LOG REPORT
// ASSGN SYS003,SYSIPT
// ASSGN SYS004,SYSLST
// ASSGN SYS001,DISK,VOL=WORK01,SHR
// TLBL SYS005,'DAILY.RCDR.BACKUP'
// DLBL SYS002,'RECORDER.CROSS.REF'
// DLBL SORTWK1,'=.SORTWK1'
// EXTENT SYS001
// EXEC TSIDARP
EXCLUDE=CICSPR16
TSIDARP199433519943655555BG
/*
/&
```

Creating the DSN/Job Cross-Reference Report (TSIDBRP)

Description

TSIDBRP generates two reports: the DSN/JOB Cross-Reference Report and the Accepted Tapes Report.

The DSN/JOB Cross-Reference Report lists tape activity and is a cross-reference to the Activity Log report created by TSIDARP.

The Accepted Tapes Report lists the controlled tapes that were accepted as input with the ACCEPT response to BIM-EPIC message EP009.

TSIDBRP uses one of three datasets as its input, depending on the UPSI value specified. For UPSI 000, 010, and 100 it uses the output dataset created by TSIDARP, called the Recorder Cross-Reference dataset. For UPSI 101, it uses either the Recorder Backup dataset created by TSIDRFS or the Recorder History dataset.

TSIDBRP uses the entire input dataset to create the contents of its reports. If you use the dataset created by TSIDARP, the time period for the reports is the same as that used for TSIDARP. If you use the Recorder Backup or History file, the time period for the reports is the time period covered by the Backup or History file.

If you use the Recorder Cross-Reference dataset as your input dataset, you may wish to run TSIDARP and TSIDBRP in the same job stream.

JCL Requirements

```
// ASSGN SYS004,SYSLST
// ASSGN SYS001,DISK,VOL=volser,SHR
// DLBL SORTWK1,'===.sortwk1'
// EXTENT SYS001
// DLBL SYS002,'recorder.cross.ref'
// DLBL SYS005,'recorder.xref.work.file'
// UPSI 000
// EXEC TSIDBRP
/*
```

or

```
// ASSGN SYS004,SYSLST
// ASSGN SYS001,DISK,VOL=volser,SHR
// DLBL SORTWK1,'===.sortwk1'
// EXTENT SYS001
// TLBL SYS005,'daily.recorder.backup'
// UPSI 101
// EXEC TSIDBRP
/*
```

UPSI Values

UPSI 000 creates both reports. It uses the Recorder Cross-Reference dataset (created by TSIDARP) as input.

UPSI 010 creates only the DSN/JOB Cross-Reference report. It uses the Recorder Cross-reference dataset as input.

UPSI 100 creates only the Accepted Tapes report. It uses the Recorder Cross-Reference dataset as input.

UPSI 101 creates only the Accepted Tapes report. It uses the Recorder Backup or History dataset as input.

Dataset Requirements

SYS002 identifies the Cross-Reference dataset created by TSIDARP. It is required with UPSI 000, 100, and 010.

SYS005 identifies the Recorder Backup or History dataset when used with UPSI 101. It also identifies the Recorder Cross-Reference work dataset when used with UPSI 000.

SORTWK1 identifies a sort work area for an internal sort. You must provide an ASSGN and a dummy EXTENT statement.

Control Statements

None

Examples

The following job creates the DSN/JOB Cross-Reference Report. It uses the Recorder Cross-Reference dataset created by TSIDARP as input.

```
// JOB TSIDBRP CROSS-REFERENCE REPORT
// ASSGN SYS004,SYSLST
// ASSGN SYS001,DISK,VOL=volser,SHR
// DLBL SORTWK1,'==.SORTWK1(N,D,P) '
// EXTENT SYS001
// DLBL SYS002,'RECORDER.CROSS.REF'
// UPSI 010
// EXEC TSIDBRP
/*
/ &
```

The following job creates the Accepted Tapes Report. It uses the Recorder File Backup as input.

```
// JOB TSIDBRP CROSS-REFERENCE REPORT
// ASSGN SYS004,SYSLST
// ASSGN SYS001,DISK,VOL=volser,SHR
// DLBL SORTWK1,'==.SORTWK1(N,D,P) '
// EXTENT SYS001
// TLBL SYS005,'DAILY.RCDR.BACKUP'
// UPSI 101
// EXEC TSIDBRP
/*
/ &
```

Job Accounting Reports by Job (TSIDJAC)

Description

TSIDJAC produces the Job Time Activity report, which reports machine use statistics by job. Statistics are detailed by job step with totals for each job. In addition, a grand total is generated for all jobs.

Use this report to find out when a particular job was run (did they really run payroll twice last night?) and what resources it used (my inventory program used 100% of my CPU?). Statistics include start and end times, duration, partition ID, CPU time, overhead, and I/O activity.

JCL Requirements

```
// ASSGN SYS004,SYSLST
// ASSGN SYS001,DISK,VOL=volser,SHR
// DLBL SORTWK1,'==.sortwk1'
// EXTENT SYS001
// TLBL SYS005,'daily.recorder.backup'
// EXEC TSIDJAC
/*
```

UPSI Values

None

Dataset Requirements

SYS005 identifies the dataset to be used as input. You can use either the Recorder Backup or History dataset.

SORTWK1 identifies a sort work area for an internal sort. You must provide an ASSGN and a dummy EXTENT statement.

Control Statements

None

Example

```
// JOB TSIDJAC JOB ACTIVITY
// ASSGN SYS004,SYSLST
// ASSGN SYS001,DISK,VOL=WORK01,SHR
// DLBL SORTWK1,'===.SORTWK1'
// EXTENT SYS001
// TLBL SYS005,'DAILY.RCDR.BACKUP'
// EXEC TSIDJAC
/*
/&
```

Job Accounting Reports by Partition (TSIDTAC)

Description

TSIDTAC produces the Date Time Activity report, which reports machine use statistics by partition and generates a grand total for machine use in all partitions.

Use this report to find out what was running in a partition at a particular time. Statistics include start and end times, duration, partition ID, CPU time, overhead, and I/O activity.

JCL Requirements

```
// ASSGN SYS004,SYSLST
// ASSGN SYS001,DISK,VOL=volser,SHR
// DLBL SORTWK1,'===.sortwk1'
// EXTENT SYS001
// TLBL SYS005,'daily.recorder.backup'
// EXEC TSIDTAC
/*
```

UPSI Values

None

Dataset Requirements

SYS005 identifies the dataset to be used as input. You can use either the Recorder Backup or the Recorder History dataset.

SORTWK1 identifies a sort work area for an internal sort. You must provide an ASSGN and a dummy EXTENT statement.

Control Statements

None

Example

```
// JOB TSIDTAC PARTITION ACTIVITY
// ASSGN SYS004,SYSLST
// ASSGN SYS001,DISK,VOL=WORK01,SHR
// DLBL SORTWK1,'===.SORTWK1'
// EXTENT SYS001
// TLBL SYS005,'DAILY.RCDR.BACKUP'
// EXEC TSIDTAC
/*
/&
```

Maintaining the Job Accounting Database (TSIDJLD)

Description

TSIDJLD creates and maintains a job accounting database.

You can access this database through CICS with the TOJA transaction. You can also develop your own reports and CICS transactions to access this data. To help you develop your own access to this data, a COBOL copy book (TSIDOJA.C) is provided.

TSIDJLD uses a VSAM KSDS to store its data. See the *BIM-EPIC Installation and System Guide*, "Initializing the Job Accounting Database" to create the database.

JCL Requirements

```
// ASSGN SYS003,SYSIPT
// ASSGN SYS004,SYSLST
// ASSGN SYS005,uuu
// ASSGN SYS001,DISK,VOL=volser,SHR
// DLBL SORTWK1,'===.sortwk1'
// EXTENT SYS001
// DLBL TSIJADB,'epicjadb.job.account',,VSAM,CAT=ucat
// TLBL SYS005,'daily.recorder.backup' (or)
// DLBL SYS005,'daily.recorder.backup'
// EXEC TSIDJLD
control cards
/*
```

UPSI Values

None

Dataset Requirements

SYS005 identifies the input dataset. You can use either the Recorder Backup or the Recorder History dataset.

TSIJADB identifies the database file that will be produced as output. This dataset is a VSAM KSDS, and must be defined prior to running TSIDJLD.

SORTWK1 identifies a sort work area for an internal sort. You must provide an ASSGN and a dummy EXTENT statement.

Control Statements

Control statements indicate what data is to be added to or deleted from the database. You can submit up to 95 control statements per execution. All fields are positional.

A control statement is required. All parameters are optional. When a field is omitted, TSIDJLD uses all relevant data.

You can submit both ADD and DEL control statements in the same execution. All deletes are performed before new information is added.

The control statement is 28 bytes long and contains five fields:

Bytes	Contents
1–3	ADD or DEL
4–11	Job name
12–21	Jobs started on or after this date
22	Blank
23–32	Jobs started on or before this date

Syntax

```
{ ADD
  DEL } jobnamemm/dd/yyyy mm/dd/yyyy
```

Parameters

Parameter	Required?	Valid	Default
Specify one of the following two (required):			
ADD		None	OFF
DEL		None	OFF
<i>jobname</i>	No	1 to 8 characters	None
<i>mm/dd/yyyy</i>	No	10 characters	None

ADD adds data from the Recorder Backup or History dataset. If ADD is the only parameter on the statement, all data are added. If other parameters are specified, only data matching those parameters are added.

DEL deletes matching data from the data base. At least one other parameter must be specified. (To delete all records from the data base, use IDCAMS to delete and redefine the database.)

jobname specifies a particular job to be added to or deleted from the data base. You can specify a generic job name by ending the name with an asterisk (for example, specify TST* to process all jobs beginning with TST). All occurrences of that job or those jobs are added or deleted. If *jobname* is omitted, all jobs on the Recorder Backup or History dataset are added or all jobs matching the dates specified in positions 12 through 28 are deleted.

mm/dd/yyyy (Positions 12 through 21) Adds or deletes jobs started on or after this date. You must use 2 digits for month and day (for example, 01/05/1997).

mm/dd/yyyy (Positions 23 through 32) Adds or deletes jobs started on or before this date. You must use 2 digits for month and day (for example, 01/05/1997).

Examples

The following job adds all information from the Recorder Backup dataset to the data base.

```
// JOB   TSIDJLD UPDATE THE JOB ACCTNG DATA BASE
// ASSGN SYS003,SYSIPT
// ASSGN SYS004,SYSLST
// ASSGN SYS001,DISK,VOL=WORK01,SHR
// TLBL  SYS005,'DAILY.RCDR.BACKUP'
// DLBL  SORTWK1,'===.SORTWK1'
// EXTENT SYS001
// DLBL  TSIJADB,'TSIJADB.JOB.ACCOUNT',,VSAM,CAT=UCAT
// EXEC  TSIDJLD,SIZE=TSIDJLD
ADD
/*
/ &
```

The following job deletes all jobs started on or before December 31, 1994 and adds all information from the Recorder Backup dataset to the database.

```
// JOB   TSIDJLD UPDATE THE JOB ACCTNG DATA BASE
// ASSGN SYS003,SYSIPT
// ASSGN SYS004,SYSLST
// ASSGN SYS001,DISK,VOL=WORK01,SHR
// TLBL  SYS005,'DAILY.RCDR.BACKUP'
// DLBL  SORTWK1,'===.SORTWK1'
// EXTENT SYS001
// DLBL  TSIJADB,'TSIJADB.JOB.ACCOUNT',,VSAM,CAT=UCAT
// EXEC  TSIDJLD,SIZE=TSIDJLD
ADD
DEL                                12/31/1997
/*
/ &
```

Online Access to the Job Accounting Database (TOJA)

Description

TOJA, a CICS transaction, gives you online access to the job accounting database created by TSIDJLD.

TOJA has four panels. The Job Selection panel is an entry panel. It allows you to select a job or a set of jobs for display. The rest of the panels display job accounting information at increasing levels of detail.

Panels

```

BIM-EPIC/VSE ONLINE JOB ACCOUNTING FACILITY

ENTER JOB NAME TO DISPLAY

ENTER (EOB) FOR ALL JOB NAMES FROM START OF FILE
ENTER PARTIAL JOBNAME FOR GENERIC DISPLAY
ENTER JOBNAME FOR SINGLE JOB DISPLAY

```

The highest level, the Job Overview panel, displays an overview of one or more jobs.

BIM-EPIC/VSE ONLINE JOB ACCOUNTING FACILITY								
.ALL								PAGE 02
JOB NAME	PHASE NAME	START DATE	START TIME	DURATION	CC	PT	AR	CPU ID
CATOJA	LNKEDT	11/17/1997	08:50:19	00:00:11	10	F4	002	444444
CATPRV	LNKEDT	11/18/1997	10:19:24	00:00:21	10	F7	002	333333
CATSDS	LNKEDT	11/18/1997	10:19:28	00:00:22	10	F8	002	333333
CICSEPIC	DTSANALS	11/18/1997	06:17:02	00:00:08	10	F2	002	222222
	IESWAITT	11/18/1997	06:17:10	00:00:14	10		000	
	DFHSIP	11/18/1997	06:17:25	02:30:32	17		000	
CICSEPIC	DTSANALS	11/18/1997	09:04:49	00:00:10	10	F2	000	222222
	IESWAITT	11/18/1997	09:05:00	00:00:04	10		000	
	DFHSIP	11/18/1997	09:05:04	00:06:37	17		000	
CICSEPIC	DTSANALS	11/18/1997	09:12:29	00:00:03	10	F2	000	222222
	IESWAITT	11/18/1997	09:12:33	00:00:01	10		000	
	DFHSIP	11/18/1997	09:12:34	00:10:39	17		000	
CICSEPIC	DTSANALS	11/18/1997	09:24:19	00:00:06	10	F2	000	222222
	IESWAITT	11/18/1997	09:24:26	00:00:02	10		000	
CICSEPI3	DTSANALS	11/17/1997	08:19:40	00:00:52	10	F2	000	444444
	IESWAITT	11/17/1997	08:20:33	00:00:53	10		000	
CICSICCF	DTSANALS	11/18/1997	06:07:49	00:00:03	10	F2	000	222222
	IESWAITT	11/18/1997	06:07:53	00:00:23	10		000	
	DFHSIP	11/18/1997	06:08:16	00:03:37	18		000	
CLEARRF	TSIDRFS	11/14/1997	10:47:16	00:00:01	10	F4	000	444444
PF3=END PF5=DETAIL PF7=BACKWARD PF8=FORWARD								

At the next level, the Program Overview panel displays an overview of the steps executed in a single job.

```

BIM-EPIC/VSE ONLINE JOB ACCOUNTING FACILITY

```

```

.ALL

```

```

PAGE 02

```

```

JOB NAME ----- CATSDS
CPUID ----- 333333
START DATE ----- 11/18/1997
START TIME ----- 10:19:28
STOP TIME ----- 10:19:50
DURATION ----- 00:00:22
CANCEL CODE (HEX) ----- 10

```

```

PHASE NAME ----- LNKEDT
PARTITION ----- F8
END DATE ----- 11/18/1997
CPU TIME ----- 00:00:01:5
OVERHEAD ----- 00:00:00:8
ALLBOUND ----- 00:00:00
PHASE CODE -- L (S=STEP L=LAST)
PHASE SEQ --- 0001

```

```

***** INPUT OUTPUT TABLES *****

```

```

400      FEC      01F      140      141      142      160
00000155 00000012 00000007 00000073 00000183 00000378 00000100
173      FEE
00000132 00000228

```

```

*****

```

```

USER INFORMATION -----
NUMBER OF AUDIT RECORDS FOR THIS PHASE -- 002
USE PF5 TO VIEW AUDIT RECORDS

```

```

PF3=END PF5=DETAIL PF7=BACKWARD PF8=FORWARD

```

Finally, the Program Audit panel displays the datasets used by a single program.

BIM-EPIC/VSE ONLINE JOB ACCOUNTING FACILITY									
FILEID	REC	CREATION	OPEN	EXPIRY	SERIAL	REEL	END	DTF	
PROCESSED	TYPE	TIME	CLOSE	CUU	DATE	ACCEPTED	TRACK	TRACK	TYPE

3F8.IJSYSLN	DO	10:19:29	C	400	11/18/1997	EPV400	030474	010000	32
3F8.IJSYSLN	DI	10:19:47	C	400	11/18/1997	EPV400	030474	010000	32
PF2=BACKWARD PF4=END									

For a detailed explanation of the fields displayed on these panels, see the Appendix.

Accessing TOJA

Step	Action
1	From CICS, enter the TOJA transaction. The Job Selection panel is displayed.
2	If you want to view all jobs, press ENTER. If you want to view several jobs beginning with the same character string, enter the string and press ENTER. If you want to view a single job, enter the job name and press ENTER.

Navigating in TOJA

To move forward in a panel, press PF8.

To move backward in a panel, press PF7.

To move to the next level of detail, place the cursor on the item you wish to display and press PF5.

To move back a level of detail, press ENTER.

Exiting TOJA

From any panel, press the CLEAR key or PF3.

12

Managing Tape Volumes

To protect controlled datasets effectively, BIM-EPIC controls the tape volumes they reside on. BIM-EPIC also protects the tape volumes it controls. Every tape that contains a BIM-EPIC-controlled dataset is recorded in the DSN Catalog. Tapes are identified by volume serial number and controlled at the version level under the name of the dataset they contain. Tape volumes are cataloged individually, as they are used for controlled datasets or initialized with TSIDINIT. BIM-EPIC provides utilities to assist you in labeling, initializing, pooling, and vaulting controlled tapes. This chapter describes BIM-EPIC tape control and tape protection utilities.

Chapter Contents

Introduction.....	12-2
Attention Routine Commands	12-5
Mounting Tapes	12-7
Tape Vaulting.....	12-17

Introduction

Tape Volume Serial Numbers

Every tape volume used to write a BIM-EPIC-controlled dataset must have a unique volume serial number. Even unlabeled tapes must be identified with their own volume serial numbers. This strict identification is essential to tape volume control.

Volume serial numbers can be any six-character combination of alphanumeric characters. Groups of volume serial numbers do not need to be consecutive; any unique volume serial number is valid. Tape pools are defined within volume serial number ranges, so it can be helpful to initialize pooled tapes with consecutive volume serial numbers.

Recognizing and Controlling Tape Volumes

When a tape volume is brought under BIM-EPIC control, BIM-EPIC gives it a special VOL1 label which identifies the volume as controlled and protects it against accidental overwrites.

BIM-EPIC looks for a controlled tape volume when it performs AVR processing. First, it looks for the special VOL1 label. If it does not find that label, it checks the DSN Catalog to see if the tape was created by CA-EPIC for CMS; if it was, the volume is accepted. If the volume fails these tests, BIM-EPIC asks the operator to specify a tape drive and to verify the volume serial number of the tape on that drive.

Unlabeled tapes do not have this special VOL1 label, so BIM-EPIC requires the operator to specify the drive that contains the volume, and to verify the volume serial number every time unlabeled processing is requested.

If an uncontrolled tape volume has a label that has expired, BIM-EPIC can use the tape, but again the operator is asked to verify the volume serial number. If that volume serial number duplicates one already in the DSN Catalog, the operator is asked to enter a new volume serial number.

HDR1 Labels and Retention

BIM-EPIC writes an expiration date of 99365 (permanent retention) on every HDR1 label created under its control. This helps to protect the tape from being overwritten outside BIM-EPIC's control (for example, when BIM-EPIC has not been started or has been deactivated). BIM-EPIC uses the DSN Catalog, not the HDR1 label, to determine each tape volume's status as active or scratch.

Scratch Tapes

Scratch tapes are produced by the SCRATCH functions of TSIDUTL, TSIDMNT, and Online Manager. The TSIDUTL "DSPLY SCRATCH" function produces a report listing all available scratch tapes. The TSIDUTL "PSCRATCH" function creates a report projecting which tapes will be eligible to be scratched on a certain date.

See the *BIM-EPIC Installation and System Guide* for information about TSIDUTL's JCL requirements. Also see the TSIDMNT SCRATCH function.

Tape Pooling

You can use tape pooling to group similar types of tapes into tape pools. For example, your high quality tapes could constitute one pool, short tapes another, long tapes a third, and so forth. You can also use pools to separate cartridge tapes from reel tapes. Datasets are assigned to tape pools through the DSN Catalog. When a pooled dataset is created, BIM-EPIC automatically requests the operator to mount a scratch tape from the appropriate pool.

Tape pools are established using the TSIDPOL macro and BIM-EPIC configuration options. Each pool is given a one character pool code (A to Z or 0 to 9) and is assigned one or more volume serial number ranges.

If a tape's volume serial number falls within a pool's ranges, the tape belongs to that pool. A pool tape can only be used to write datasets assigned to that pool. If a volume serial number does not fall in any of the defined ranges, the tape is not a pool tape. It can only be used to write datasets that are not assigned to a tape pool.

Datasets can be assigned to tape pools through the Online Manager, the TSIDMNT program, or the use of the `OPTION=TPL=x` parameter on a DD statement.

Tape Ownership

The tape ownership feature provides another form of tape pooling. An owner ID is a two character code that is used as the first two characters in a tape's volume serial number.

When a dataset is cataloged with an owner ID, it can only be written on tapes with volume serial numbers which begin with the code specified.

Tape ownership is different from tape pooling because a pool tape can only be used for datasets assigned to that pool. Tape ownership does not restrict the use of tapes in that way. Only the dataset is restricted. The tape can be used for any dataset.

Specifying Tape Size

You can restrict datasets to small or large tapes with the size specification feature.

Small tapes are distinguished by their volume serial numbers, which must end with S. Large tapes have no restrictions to their volume serial numbers. You can designate that a dataset should be written to small tapes with online or manual catalog maintenance, using the Tape Size field.

Revolving Tape Datasets

In some cases, you may want to use a specific set of tapes for a dataset, such that all versions of that dataset are contained on that set of tapes only.

The TSIDMNT CATALOG command contains a parameter (REV) that allows you to define revolving datasets.

Specifying `REV=YES` for a dataset establishes an exclusive relationship between the dataset and the tapes it uses: a revolving tape can only be used to write its own dataset, and a revolving tape dataset can only be written on its own tapes.

If the dataset is new or it does not own enough tapes, new tapes can be added with the ADD command and then scratched with the SCRATCH command (using Online Manager or the TSIDMNT program).

Multi-Dataset/Multi-Volume Processing

BIM-EPIC supports multi-dataset, multi-volume, and multi-dataset/multi-volume processing. A tape volume that contains a dataset that is a member of a multi-dataset group cannot be scratched until all datasets in the group have been scratched.

Work Tapes

A work tape dataset is one that has been cataloged with the attribute WRK=YES or defined with DISP=(NEW,DELETE) on a DD statement. A work tape dataset can be written on any available scratch tape. The data on a work tape is not usually saved after the end of the job in which it was created, so there is usually no need to place an external dataset label on a work tape. For this reason, the EP005 message (**LABEL**) is not issued for work tapes unless special option EPS043=YES.

Assigning Tape Drives

BIM-EPIC uses a process called **Automatic Volume Recognition (AVR)** to automatically search available tape drives for a required tape volume whenever a program issues an OPEN request. When AVR locates the required volume, BIM-EPIC makes the necessary assignment. When the program issues a CLOSE request, BIM-EPIC releases the assignment and makes the drive available to other programs unless directed otherwise by a parameter on the TLBL or DD statement.

It is the operator's job to ensure that tape drives are available to BIM-EPIC when it is searching for a tape. A tape drive is available if it:

- Contains a valid tape (the volume serial number is in the DSN Catalog, or the tape has been initialized by the TSIDINIT or DITTO program)
- Is not in the “device down” status
- Is not attached to any virtual machine (if Autoattach is on), or is attached to the machine running the job (if Autoattach is off)
- Is not assigned to any partition
- Is ready and at load point
- Supports the mode setting selected for the dataset

Under most circumstances, you do not have to make drive assignments with the VSE ASSGN statement. If Autoattach is on, you should avoid using the VM/CP ATTACH or DETACH commands to control the tape drives. Instead, DETACH every tape drive and let BIM-EPIC do the work for you.

If Autoattach is on and you do use the ATTACH command, remember to issue the VM/CP DETACH command for the tape drive when you are finished, so that BIM-EPIC can use the drive in another virtual machine.

BIM-EPIC only uses AVR for controlled tapes. If the required tape is not a BIM-EPIC-controlled tape, the operator will be asked to specify the address of the tape drive to be used.

Attention Routine Commands

Description

Certain BIM-EPIC services can be invoked by commands that the operator issues to the VSE Attention Routine (AR).

Command Summary

The BIM-EPIC Attention Routine commands are:

Command	Action
EP ? (or EP HELP)	Displays help information for BIM-EPIC AR commands.
EP END ST $name$	Terminates a BIM-EPIC Dispatcher subtask. <i>This command should be used only when directed by BIM-EPIC Technical Support.</i> $name$ is the name of the subtask being ended. ACL is a valid subtask name.
EP END TASK	Ends BIM-EPIC Dispatcher processing.
EP REW, cuu	Rewinds the selected tape drive.
EP RUN, cuu	Unloads the selected tape drive.
EP STATUS[, cuu]	Displays status of all accessible tape drives, or of a particular tape drive.
EP TRQ CANCEL n	Deletes a work request that has not been processed by the BIM-EPIC Dispatcher. n is the identification number of the request to be deleted. It is obtained using the EP TSTAT command.
EP TSTAT	Displays any work requests waiting to be processed by the BIM-EPIC Dispatcher.

How to Enter the Commands

BIM-EPIC Attention Routine commands must begin with “EP” followed by a space. For example, to unload the tape on drive 881, enter:

```
EP RUN,881
```

EP930 Message

The message EP930 UNABLE TO OBTAIN LOCK - COMMAND BYPASSED may be issued on some systems with heavy activity (particularly when two physical CPU's are active with SHARE=YES). The Attention Routine cannot afford to wait for a successful lock, so BIM-EPIC issues this message when it cannot obtain one immediately. Try the command again.

FREE and STATUS Commands

Description

In addition to providing status information using the AR command facility, BIM-EPIC also allows the operator to respond FREE or STATUS to many BIM-EPIC messages.

The FREE command lists all tape drives that are:

- In READY status or with no tape mounted
- Not assigned to any partition
- Not attached to any other machine

The STATUS command lists the current status of each tape drive.

Mounting Tapes

Output

Description

BIM-EPIC controls which tapes can be used for output. It checks the volume serial number of every tape processed. If the tape mounted is controlled, its volume serial number is checked with its corresponding DSN Catalog entry to ensure that it is in scratch status before it can be used as output.

If the tape is uncontrolled, BIM-EPIC checks to see if the tape's volume serial number exists in the DSN Catalog. If it does not yet exist, the tape is cataloged and used automatically. If the volume serial number already exists in the catalog, BIM-EPIC issues a message to allow the operator to enter a new volume serial number. You do not have to pre-initialize all your tapes to bring them under BIM-EPIC's control.

The configuration option UNCTAP affects uncontrolled tape handling.

Input

Requesting the Previous Version

To request the previous version of a dataset, type RERUN in response to BIM-EPIC's request for a tape mount. A new tape mount request will then be issued for the previous version.

Example

If a tape mount request is issued for version 1 and you reply RERUN, a new request is issued for version 2. You can reply RERUN as many times as there are versions for the dataset.

Requesting a Specific Version

To request a specific version, type RERUN,n in response to a request for a tape mount, where *n* is the number of the version requested.

Example

To issue a new tape mount request for version number 3, type RERUN,3.

Using ACCEPT

The operator can be given the option to reply ACCEPT to a mount request to force BIM-EPIC to use any desired volume. Each time ACCEPT is used, a notation is made in the Accepted Tapes Report generated by TSIDBRP. The use of the ACCEPT reply is governed by the ACCEPT configuration option.

WARNING!

ACCEPT=YES disables BIM-EPIC's supervision of the input process and places responsibility for tape selection in the hands of the operator.

Initializing Tapes (TSIDINIT)

Description

TSIDINIT initializes tapes with BIM-EPIC VOL1 labels and enters them in the DSN Catalog under the dataset name '***.VOLUME.DSN.**'. The tapes are then ready for use as controlled scratch tapes.

Up to ten tape drives can be used in the initialization process.

TSIDINIT can be executed from JCL or from the operator console. If it is executed from the console, the operator is queried for all processing information.

To execute from the console, enter:

EXEC TSIDINIT

after any VSE '// PAUSE' statement.

JCL Requirements

```
[// UPSI xxxx]
// EXEC TSIDINIT [,PARM='CONSOLE' ]
           or
// EXEC TSIDINIT [,PARM='ACL=aclname[,COUNT=nnnnn]']

           cuu statement
           volser statements
/*
```

UPSI Values

To specify...	Use UPSI setting...
Sequential initialization	1xxx
Random initialization	0xxx
Forced initialization	x1xx
Scratch tape reinitialization	xxx1
Inclusion of the CUU statement	xx1x
Use an ACL for Initialization	xx101

Types of Initialization

When more than one tape is to be initialized, you can specify each tape's volume serial number in individual control statements, and the statements can be placed in any order. This is referred to as **random initialization**. You can also specify the starting volume serial number, and TSIDINIT will initialize successive tapes in ascending volume serial number sequence. This is referred to as **sequential initialization**. In sequential initialization, TSIDINIT continues to initialize tapes until the operator replies STOP to the mount message.

Sequential initialization requires that the last four characters of the volume serial number be numeric (for example, EP1200 or A67019). Random initialization places no restrictions on the volume serial number.

TSIDINIT normally checks each tape's actual volume serial number before writing in the tape to ensure that an active tape is not initialized by mistake. It also protects against the creation of duplicate volume serial numbers. If one of the volume serial numbers specified already exists in the DSN Catalog, BIM-EPIC issues an error message and request a new volume serial number.

Forced Initialization: Use this only if initializing brand new tapes that have never been previously initialized. Forced initialization directs TSIDINIT to bypass any attempt to read data from the volume and to immediately initialize the volume with the requested volume serial number. *Caution should be exercised with forced initialization. It bypasses tape label checking, so active data can be lost if the wrong tape is mounted. If the volume serial number already exists in the DSN Catalog, TSIDINIT issues an error message, and the tape will have nothing but tape marks on it.*

If it is necessary to reinitialize a tape that is currently in scratch status, execute TSIDINIT with UPSI xxx1.

If the initialization process is to use an ACL:

PARM='CONSOLE' cannot be used. CUU and volume serial information must be submitted via SYSIPT statements.

PARM='ACL=aclname' or PARM='ACL=aclname,COUNT=nnnnn' must be present on the // EXEC statement.

Random, sequential, or forced initializations are permitted. Forced should be used with caution. If a volser should accidentally be specified for an existing volser containing an active dataset, or if the volser is owned by another operating system or machine, EPIC will initialize it and its data will be lost.

In multiple ACL's are present, only one of them can be used in a TSIDINIT execution. If concurrent TSIDINIT executions are to be done, each should specify a different ACL Name.

Since the ACL mounts volumes by reading their barcodes, the volume serial numbers specified to TSIDINIT must match the appropriate barcodes.

PARM Values

PARM='CONSOLE' can be specified on the VSE EXEC statement to allow the operator to control initialization. This has exactly the same effect as entering EXEC TSIDINIT after a VSE '// PAUSE' statement.

PARM='ACL=*aclname*[,COUNT=*nnnnn*]'

The parameters are positional – ACL= must precede COUNT=

aclname designates the ACL to use for initialization. It must be same as the ACLNAME parameter used in defining the ACL's cuu's in the EPIC Resource Dataset (ERD).

count designates the number of volumes to initialize. It is only used in **sequential** initializations and is ignored if present for **random** initializations. A maximum of 99,999 volumes can be initialized in a TSIDINIT execution. In deciding on the number of volumes to initialize in one execution, consider the time involved in robotic mount/demount activity.

CUU Statement

Normally, the operator is asked to specify the tape drives to be used. If specific tape drives are to be used, operator intervention can be eliminated by using a CUU statement. Up to ten drives can be specified.

If an ACL is being used, there will not be any operator intervention. After a volume is initialized, it is unloaded via a RUN command. Once the RUN completes and the robot begins moving to do the "return to slot" action, a mount cannot take place until this return action has completed. For this reason, it is recommended that two or three, but no more, CUU's be used during a TSIDINIT execution.

User must insure that the ACL cuu's specified in the CUU statements are not in use when TSIDINIT is submitted for execution.

If in a VM environment, the cuu's must not be attached to a guest machine other than the one in which TSIDINIT will execute. If unattached, TSIDINIT will attach the cuu's when it begins execution and detach them when finished executing.

The format of the CUU statement is as follows:

Syntax

CUU=*cuu*,*cuu*,*cuu*, . . .

The CUU statement requires the use of UPSI xx1x in your JCL.

VOLSER Statements

VOLSER statements identify which volume serial numbers are written in the VOL1 label.

If random initialization is requested, one VOLSER statement is required for each volume serial number. If sequential initialization is requested, use only one statement to specify the starting volume serial number.

Examples

In this example, TSIDINIT performs sequential initialization of uncontrolled tapes that have labels or tape marks. It initializes the first tape with the volume serial number EP0001, the second tape with EP0002, the third tape with EP0003, and so on until the operator stops the process. It does label checking to ensure that the tape does not contain an active dataset. It also checks the DSN Catalog to ensure that there is no duplication of volume serial number before it initializes each tape.

```
// JOB INITTAPE
// UPSI 1000
// EXEC TSIDINIT
    EP0001
/*
/ &
```

In this example, TSIDINIT performs random initialization of uncontrolled tapes that do not have labels or tape marks. It initializes the first tape with the volume serial number PR0001, the second tape with PR0024, the third tape with PR0047. It does not prevent duplication of volume serial number and does not do label checking to ensure that the tapes mounted are valid scratch tapes.

```
// JOB INITTAPE
// UPSI 0100
// EXEC TSIDINIT
    PY0001
    PY0024
    PY0047
/*
/ &
```

In this example, TSIDINIT uses an ACL to do sequential initialization of tapes not in the EPIC Catalog that have labels or tape marks. It initializes the first tape with volume serial number SR0010, the second with SR0011, and so on until 10 volumes have been initialized. (SR0010 through SR0019)

```
// JOB INITTAPE
// UPSI 10101
// EXEC TSIDINIT,PARM='ACL=B1VTS,COUNT=10'
    CUU=480,481
    SR0010
/*
/ &
```

In this example, TSIDINIT uses an ACL to do random initialization of tapes not in the EPIC Catalog that have labels or tape marks. It initializes the first tape with volume serial number JCSR09, the second with WXY981, and so on until all volume serial number specifications have been read.

```
// JOB INITTAPE
// UPSI 00101
// EXEC TSIDINIT,PARM='ACL=A2ATL'
    CUU=480,481
    JCSR09
    WXY981
    ...
    ...
/*
/ &
```

In this example, TSIDINIT uses an ACL to do random initialization of tapes not in the EPIC Catalog that do not have labels or tape marks, i.e. *forced* initialization. It initializes the first tape with volume serial number JCSR09, the second with WXY981, and so on until all volume serial number specifications have been read.

```
// JOB INITTAPE
// UPSI 01101
// EXEC TSIDINIT,PARM='ACL=A2ATL'
    CUU=480,481
    JCSR09
    WXY981
    ...
    ...
/*
/ &
```

Printing External Volume Serial Labels (TSIDLBL)

Description TSIDLBL creates external volume serial number labels in large block characters. These labels contain the volume serial number and a short optional comment.

Standard spacing for each label is 1 line for the comment, 7 lines for the volume serial number, and two blank lines between labels.

Example

```
THIRTY CHARACTER COMMENT
*****          *****          *****          *****          *
*                *  *              *  *              *  *              *  *
*                *  *              *  *              *  *              *
*****          *****          *  *              *  *              *
*                *                *  *              *  *              *
*                *                *  *              *  *              *
*****          *                *****          *****          *****          *
```

JCL Requirements

```
    [// UPSI 1]
// EXEC TSIDLBL
control statements
/*
```

UPSI Values

UPSI 1 causes the printer to skip to channel 1 (top-of-form) after each label. This allows you to use labels of any size. When UPSI 1 is used, TSIDLBL assumes that the printer has a correctly loaded forms control buffer or carriage control tape that defines the size of the labels.

Syntax

```
volser[-volser] [ comment ]
```

Parameters

Parameter	Required?	Valid Entry	Default
<i>volser-volser</i>	Yes	1 to 6 characters	None
<i>comment</i>	No	1 to 30 characters	None

volser [-volser] specifies the range of volume serial numbers for the labels you want to create. If only one label is to be created, omit the second *volser*. The last four characters in the volume serial number must be numeric.

comment specifies the comment to be printed on each of the labels.

Example

In this example, EP0001-EP0100 DSN CONTROLLED indicates that one label is to be created for each volume serial number from EP0001 to EP0100, inclusive. Each label in this series will contain the comment 'DSN CONTROLLED'.

PR0001-PR0050 PAYROLL indicates that one label is to be created for each volume serial number from PR0001-PR0050, inclusive. Each label in this series will contain the comment 'PAYROLL'.

```
// JOB PRINTLBL
// EXEC TSIDLBL
EP0001-EP0100 DSN CONTROLLED
PR0001-PR0050 PAYROLL
/*
/&
```

Printing Dataset Labels (TSIDBPR and TSIDPRT)

Description

With BIM-EPIC, you can create gummed dataset labels automatically when datasets are opened for output.

You can create labels in a batch partition or through CICS. The TSIDBPR program creates labels in a batch partition. The TSIDPRT transaction creates labels through CICS.

Automatic label creation requires you to write a subroutine. For installation and subroutine requirements, see the *BIM-EPIC Installation and System Guide*. Dataset label creation is enabled using the RMNTPRT configuration option.

Label Creation in a Batch Partition

Execute TSIDBPR in any partition. Be sure to use a partition which is not needed for other jobs, since TSIDBPR remains active in that partition until it is deactivated.

To deactivate TSIDBPR, execute TSIDPEN in any partition. This terminates TSIDBPR and releases the label printer.

Label Creation in CICS

TSIDPRT can be activated automatically through the CICS PLTPI table. If it is activated using the PLTPI, enter the TPRT transaction in CICS. When you activate TSIDPRT this way, the terminal you use is not released until the TPRT is deactivated or until CICS is terminated.

To deactivate TPRT, execute TSIDPEN in any available partition. This releases the label printer and any terminal used by TPRT.

Tape Vaulting

Introduction

Description

BIM-EPIC's vaulting subsystem controls tape vaulting automatically. This section presents an overview of the programs and procedures involved in tape vaulting.

Components

Vault locations are established using the VAULTDEF macro, which defines the name and, if necessary, the size of each vault and the number of slots it contains. Vaults are identified by integers. The main library is always Vault 0 (zero). BIM-EPIC assigns the other vault numbers to each vault in the sequence defined by the VAULTDEF macro.

Vaulting methods are established using the METHOD macro. Each method defines a distinct sequence of vault locations for all versions of a dataset vaulted with that method.

The VAULTDEF and METHOD macros are documented in the *BIM-EPIC Installation and System Guide*.

How Vaulting Works

Datasets are designated for vaulting using the DSN Catalog. If a dataset has a vaulting method assigned to it, its versions are vaulted automatically whenever the vault movement program (TSIDVLT) is run.

TSIDVLT moves cataloged tape volumes to their new vault locations (and slot numbers if defined). In addition, TSIDVLT creates two vault movement reports. Each report lists only the tapes which need to be moved, sorted by vault location. These reports are:

Report	displays...
Picking List	Tapes listed by their current vault and slot location and indicating the vault and slot destination to which each tape is to be moved.
Receiving List	Tapes listed by their newly assigned vault and slot destinations and indicating the current vault and slot where each tape can be found.

Vaulting ACL Cartridges

If an Automated Cartridge Library (ACL) is supported, TSIDVLT also writes the volume serial numbers of all cartridges moved from Vault 0, in the ACL Inventory Management Library. At a later time, these volumes can be ejected from the ACL using the EP ACL EJECT command or the TSIDUTL program's ACLEJECT control statement. See the *BIM-EPIC Installation and System Guide* for more information on these commands.

Additional Reports

Reports listing current vault inventories can be created using TSIDUTL. TSIDPRV creates a report that cross references datasets with the vaulting methods assigned to them. See the *BIM-EPIC Installation and System Guide* for more information on TSIDUTL.

Assigning Vaulting Methods to Datasets

Description

After vaults and vaulting methods have been defined, they must be assigned to datasets before automatic vaulting can occur. This is done using the DSN Catalog.

Vaulting methods are assigned to datasets using the CATALOG or UPDATE command (in Online Manager or TSIDMNT).

In Online Manager, place the vault method number in the Vault Method field of the EDD.

In TSIDMNT, use the VLT parameter to supply the vault method number to the EDD.

Moving Volumes (TSIDVLT)

Description

TSIDVLT should be executed every day. TSIDVLT moves tapes from one vault to another and creates movement reports. TSIDVLT can be instructed to:

- Perform actual vaulting actions. Movement from one vault/slot to another occurs and a report describing the activity is produced.
- Perform projected vaulting actions. A report is produced describing what movements would take place, but no actual vault/slot movement occurs.

JCL Requirements

```
// DLBL SORTWK1...
// EXTENT SYS001
// EXEC TSIDVLT
control statements
```

Control Statements

Four input control statements are provided to tell TSIDVLT what action to take. The control statements are read from SYSIPT and can begin in any column. Only one control statement can be used in any execution of TSIDVLT. A control statement must be present; there is no default.

Control Statement	Function
MOVESLT	Do vault movement and produce a report sorted by vault and slot number.
MOVESER	Do vault movement and produce a report sorted by volume serial number.
PROJSLT	Do not do vault movement, but produce a report of what movement would occur, sorted by vault and slot number.
PROJSER	Do not do vault movement, but produce a report of what movement would occur, sorted by volume serial number.

Dataset Requirements

SORTWK1 is required.

Example

```
// JOB VAULTMOV
// DLBL SORTWK1, '===.SORTWK1,(D) '
// EXTENT SYS001
// EXEC TSIDVLT
MOVESLT
/*
/&
```

Reporting Current Locations of Vaulted Tapes

Description

The DSPLY function of TSIDUTL creates reports which provide vault information for tape volumes. These reports can select and display tape datasets by vault.

Parameter	Action	Example
ALL	Lists vault information for all tape datasets.	DSPLY ALL,SEQ=DSN
VAULT	Lists the tapes contained in every vault except Vault 0 (main library). When SEQ=VAULT is specified, tapes are listed in vault and slot sequence and a page break occurs after each vault.	DSPLY VAULT,SEQ=VAULT
VLT	Lists the contents of a single vault.	DSPLY ALL,VLT=2,SEQ=DSN

Manually Controlled Vaults

Description

A **manually controlled vault** is any vault whose number is greater than the value of the AVAULTS parameter in the VAULTDEF macro. Since manually controlled vaults are not known to BIM-EPIC, they must be tracked manually.

Any version of any dataset can be moved to a manually controlled vault. Tapes are moved to or from manually controlled vaults using the MOVE function of Online Manager or TSIDMNT.

WARNING!

If a dataset is under the control of an automatic vaulting method, TSIDVLT executions override any manual vaulting.

Vault To Dataset Cross-Reference Report (TSIDPRV)

Description	TSIDPRV produces the Vault Method Report. This report lists the vaulting method for each dataset, in dataset name sequence. For a dataset to appear on this report, it must have a vaulting method assigned to it in the DSN Catalog.
--------------------	---

JCL Requirements	<pre>// DLBL SORTWK1... // EXTENT SYS001 // EXEC TSIDPRV</pre>
-------------------------	--

UPSI Values	None
--------------------	------

Dataset Requirements	SORTWK1 is required.
-----------------------------	----------------------

Example	<pre>// JOB VLTSMETH // DLBL SORTWK1'===.SORTWK1,(D)' // EXTENT SYS001 // EXEC TSIDPRV /&</pre>
----------------	---

Dataset to Tape Report (TSIDDIS)

Description

TSIDDIS produces a summary report correlating dataset versions and tape volume serial numbers. It is listed in dataset name order. There is one entry for each dataset version. If a version spans more than one volume, there is an entry for each volume. See the Appendix for an example of this report.

JCL Requirements

```
// DLBL SORTWK1  
// EXEC TSIDDIS
```

UPSI Values

None

**Dataset
Requirements**

SORTWK1 is required.

Example

```
// JOB TAPE DATASET LISTING  
// DLBL SORTWK1, '==.SORTWK1'  
// EXTENT SYS001  
// EXEC TSIDDIS  
/&
```

Tape to Dataset Report (TSIDSDS)

Description	TSIDSDS produces a summary report correlating dataset versions and tape volume serial numbers. It is listed in volume serial number order. There is at least one entry for each version of a dataset. If a version spans more than one volume, there is an entry for each volume. See the Appendix for an example of this report.
--------------------	---

JCL Requirements	// DLBL SORTWK1 // EXEC TSIDSDS
-------------------------	------------------------------------

UPSI Values	None
--------------------	------

Dataset Requirements	SORTWK1 is required.
-----------------------------	----------------------

Example	// JOB TAPE VOLSER LISTING // DLBL SORTWK1, '==.SORTWK1' // EXTENT SYS001 // EXEC TSIDSDS /&
----------------	--

13

Disk Space Management

This chapter discusses how BIM-EPIC can help you to maximize and manage your disk space.

Chapter Contents

Introduction	13-2
Dataset Fragmentation	13-2
Secondary Allocation	13-3
Truncation	13-4
Converting Uncontrolled Disk Datasets	13-5
Placing Datasets on Specific Volumes in a Pool	13-6
VTOC Utilities	13-7
Introduction	13-7
VTOC Listings (TSIDLVT)	13-8
Online VTOC Displays (TVTO)	13-11
Offload System	13-15
Introduction	13-15
Selecting Datasets To Be Offloaded (TSIDOFD)	13-17
Offloading and Backing Up Datasets (TSIDOFB)	13-22
Onloading and Restoring Datasets (TSIDONL)	13-27
TSIDONL Control Statements	13-29
Differences Between TSIDONL Onload and TSIDONL Restore	13-32
Consolidating Offload Versions (TSIDOFB)	13-34

Introduction

Dataset Fragmentation

Description

When you create a dataset with a large primary allocation, there may not be a large enough free space in the pool to contain the entire primary extent. BIM-EPIC can respond to this condition by allocating the largest contiguous free space to the primary extent and allocating secondary extents when the primary extent becomes full. This process is called **dataset fragmentation**.

Fragmentation can be enabled globally by specifying the configuration option FRAG=YES. Fragmentation can be enabled for an individual dataset by:

- Specifying the “F” option on a DLBL statement
- Specifying the MXIG or ALX subparameter in the SPACE parameter on a DD statement
- Allowing the operator to respond FRAG to the EP038 message

Secondary Allocation

Description

BIM-EPIC automatically allocates secondary extents to eligible datasets when the primary extent has been filled.

Secondary allocation is not available for direct access or TYPEFLE=WORK datasets. Secondary allocation is available for SORTWK datasets only if the SORT program specifically allows it.

For cataloged datasets, you can specify the size of the secondary extents and the maximum number of secondary extents (up to 255). If you use DD statements for your disk datasets, the size of the secondary extents can be specified in the SPACE parameter, and the maximum number of secondary extents can be specified in the VOLUME parameter. If you do not define a dataset's secondary allocation requirements in one of these ways, BIM-EPIC automatically allocates up to 255 secondary extents, and each extent will be one-half the size of the primary extent.

Truncation

Description

Any allocated extent that was not completely filled can be truncated so that the unused portion of the extent is freed. BIM-EPIC performs truncation for controlled datasets by altering both the VTOC and the DSN Catalog to show only the space actually used. Extents are always truncated on track or control interval boundaries.

Truncation occurs only on output for sequential disk datasets. It should be inhibited for TYPEFLE=WORK DTFs and for datasets that are opened as sequential I/O but are not accessed sequentially (for example, DL/1 datasets). Truncation is automatic for datasets that you define with DLBL statements. It must be requested using the RLSE subparameter of the SPACE parameter when using BIM-EPIC DD statements.

Truncation can also be inhibited by the DLBL option “N”.

Converting Uncontrolled Disk Datasets

Description

Existing uncontrolled disk datasets can be placed under BIM-EPIC control. Any existing dataset can be controlled, as long as all its extents reside in a single defined disk pool. You can convert a disk dataset to catalog control or Start Track 1 control.

To Convert to Catalog Control

Step	Action
1	Catalog the dataset name as a work file using the Online Manager (specify “WRK” on the Define an EDD panel) or the TSIDMNT CATALOG command (specify the WRK=YES parameter).
2	Renumber the dataset's extents using the TSIDVUT RENUM function. BIM-EPIC-controlled disk files must begin with extent sequence number 1.
3	Change the dataset's VTOC expiration date to permanent using the TSIDVUT RETAIN function. Catalog-controlled disk files are scratched according to the retention criteria specified in the EDD, not the VTOC expiration date.
4	Add the version to the DSN catalog using the Online Manager or the TSIDMNT ADD command. Be sure to specify the version block size, if it is known.
5	If generation retention for the dataset has been specified, update the file type from WRK to GEN using the Online Manager or TSIDMNT UPDATE command <i>after</i> the existing version has been added to the DSN catalog.

To Convert to Start Track 1

Step	Action
1	Renumber the dataset's extents using the TSIDVUT RENUM function. BIM-EPIC-controlled disk files must begin with extent sequence number 1.
2	If necessary, change the dataset's VTOC expiration date using the TSIDVUT RETAIN function. Expired files in a BIM-EPIC disk pool are eligible for immediate reallocation.
3	Change the dataset's EXTENT statements to specify a starting track of 1, or whatever number was specified in the STRTRK configuration option.

Example

```
// EXTENT SYS001,POOL50,1,0,1,10
```

Placing Datasets on Specific Volumes in a Pool

Description

All disk datasets can be assigned to a BIM-EPIC pool. If your pool is made up of more than one volume, the dataset can be placed on any of those volumes. If you wish, you can place datasets on a specific volume.

Procedure

Step	Action
1	Quiesce all other volumes in the pool.
2	Create the dataset and inhibit truncation.
3	Reuse the same space when re-creating the dataset. See 7-10 in this manual for further information.

VTOC Utilities

Introduction

Description

BIM-EPIC provides the following VTOC utilities: TSIDLVT, TVTO, and TSIDVUT.

Utility	Description
TSIDLVT	Displays VTOCs by BIM-EPIC pool or by volume, and in extent or dataset sequence. Can display both active and expired datasets, or active datasets only.
TVTO	Online utility. Displays VTOCs by volume only.
TSIDVUT	Allows modification of existing entries. Deletes active datasets, retains a dataset permanently, renames a dataset, or converts an uncontrolled dataset to BIM-EPIC control. See the <i>BIM-EPIC Installation and System Guide</i> for information.

WARNING!

Use TSIDVUT to update the entries of uncataloged datasets only.

Printing VTOC Entries (TSIDLVT)

Description	<p>TSIDLVT produces printed reports of VTOC entries. It can display entries for an entire BIM-EPIC pool or a specific volume.</p> <p>TSIDLVT prints expired datasets, unexpired datasets, and free space. Extents are shown in track/block format and dates in MM/DD/YYYY format. Control statements can be used to change the display format.</p> <p>At the end of each report, TSIDLVT prints the total free space in the pool or volume. This total includes expired extents as well as unused space.</p>
JCL Requirements	<pre>// DLBL SORTWK1,... // EXTENT SYS001 // EXEC TSIDLVT control statements /*</pre>
UPSI Values	<p>None</p>
Dataset Requirements	<p>SORTWK1 is required for all functions.</p>
Control Statements	<p>You must include one control statement per report. The control statement must specify the volume or pool to be displayed.</p> <p>You can use control statement parameters to customize the report.</p> <p>The EXP, UNEXP, and FREE parameters limit the contents of your report.</p> <p>For example, if your control statement specifies:</p> <pre>SER=SYSWK1,UNEXP</pre> <p>the report lists only unexpired datasets.</p> <p>To receive a report containing only expired datasets and free space, use a control statement like this one:</p> <pre>SER=SYSWK1,EXP,FREE</pre> <p>If none of these parameters are specified, all three types of data will appear on the report.</p>
Control Statement Parameters	<div><div><div><div><div></div><div><i>SER=volser</i></div></div><div><div><i>poolname</i></div><div></div></div></div></div></div>


```
[ ,EXP ]
[ ,UNEXP ]
[ ,FREE ]
[ ,DSN ]
[ ,INPOOL ]
[ ,EURO ]
[ ,CCHH ]
```

Parameters

Parameter	Required?	Valid	Default
One of the following two is required:			
<i>volser</i>		1 to 6 characters	None
<i>poolname</i>		1 to 6 characters	None
ALL	No		ON
UNEXP	No		OFF
EXP	No		OFF
FREE	No		OFF
DSN	No		OFF
INPOOL	No		OFF
EURO	No		OFF
CHH	No		OFF

You must specify either `SER=volser` or `poolname`.

volser specifies the disk volume serial number to report.

poolname specifies the name of the BIM-EPIC disk pool to report.

EXP prints expired datasets.

UNEXP prints unexpired datasets.

FREE prints free space.

DSN prints in dataset name sequence. It is ignored if FREE is also specified.

INPOOL lists extents within pool limits only. If INPOOL is not specified, TSIDLVT ignores pool limits on a volume.

EURO prints dates in European format (dd/mm/yyyy). By default dates are printed in USA format (mm/dd/yyyy).

CCHH prints extents in cylinder/head (cchh) format. If cchh is not specified, extents will be shown in relative track/block format.

Examples

The following job creates a report of the VTOC of volume SYSWK1.

```
// JOB VTOC
// DLBL SORTWK1, '===.SORTWK1,(X) '
// EXTENT SYS001
// EXEC TSIDLVT
    SER=SYSWK1
/*
/ &
```

The following job creates two reports. The first one, reporting POOL50, prints the standard report. The second, reporting POLFBA, lists only unexpired and expired datasets. They are listed in dataset name sequence.

```
// JOB VTOC
// DLBL SORTWK1, '===.SORTWK1,(X) '
// EXTENT SYS001
// EXEC TSIDLVT
    POOL50
    POLFBA, UNEXP, EXP, DSN
/*
/ &
```

Online VTOC Displays (TVTO)

Description

The TVTO CICS transaction displays information about entries found in the VTOC on a specific DASD volume.

TVTO is a pseudo-conversational transaction, can only be utilized on an 80-character wide screen, will utilize 31-bit storage for the majority of its work areas (if available), and will automatically use the alternate screen size if larger than the default screen size.

Syntax

TVTO (<=== transfers to HELP screen)
or
TVTO cuu (<=== transfers to VTOC screen)
or
TVTO volser (<=== transfers to VTOC screen)

Help Screen

```
CUU ??? <== Enter CUU or set cursor at desired entry (PF3=END)      Pool: ALL
Name|Area|Space|Cdate|Edate: AREA   Cyl|Trk: TRK   High|Size: SIZE   Expire: YES
140=DOSRES
141=SYSWK1
142=FBAA001
160=CKD001
161=CKDP11
162=CKDP12
163=CKDP13
164=CKDP14
165=CKDP15
172=VSEPWR
173=VSELBL
174=VSEPAG
178=VSEPW2
179=FBAP11
17A=FBAP12
17B=FBAP13
17C=FBAP14
17D=FBAP15
400=EPV400
402=VDI402
```

The HELP screen is displayed when only the TVTO transaction-id is entered from a blank CICS screen, whenever an invalid request is attempted, or by pressing the PF1 key from the VTOC screen. Error and informational messages are displayed on the top screen line just to the right of the cuu field. Pressing PF3 from the HELP screen will terminate the TVTO transaction and return control to CICS with a blank screen.

The right-most field on screen line 1 and screen line 2 display all the current settings which control the manner and sequence for entries to be displayed on the VTOC screen.

Pool: controls which DASD entries will be displayed on the Help Screen and controls whether disk pool information (extracted from the current TSIDPOL phase) will be included in the VTOC display. Note: disk pool information will be included in the VTOC display only when an AREA or NAME sort sequence is chosen.

ALL Indicates that only DASD entries that are a member of *any* disk pool are to be displayed on the Help Screen and to include both the lower

and upper limits for *every* disk pool containing space on the requested volume on the VTOC Screen (may also be specified as YES, Y, or A).

- NONE (Default) Ignore all disk pool information. All DASD entries defined to this VSE system will be displayed on the Help Screen and no pool limits will be displayed on the VTOC Screen (may also be specified as NO or N).
- Poolname Include only disk pool information for this Poolname. Only DASD entries defined to this VSE system that are also members of this Poolname will be displayed on the HELP Screen and only the lower and upper limits for this Poolname will be displayed on the VTOC Screen

Name|Area|Space|Cdate|Edate: controls the VTOC screen sort sequence (only the first letter of each choice need be keyed):

- Name Entries are sorted by File name.
 Area (Default) Entries are sorted by starting location.
 Space Entries are sorted by size (space occupied).
 Cdate Entries are sorted by creation date.
 Edate Entries are sorted by expiration date.

Cyl|Trk: controls the display of disk addresses for CKD devices only (only the first letter of each choice need be keyed):

- Cyl: (Default with EPS065=NO) Disk addresses are displayed in cylinder-head format.
 Trk: (Default with EPS065=YES) Disk addresses are displayed in track format.

High|Size: controls the display to reflect either high disk address or size occupied in addition to the starting (or low) disk address. High is incompatible when requesting the VTOC screen sort sequence of Space and will be automatically reset to Size when that screen sort sequence is requested with High. (Only the first letter of each choice need be keyed):

- High Show high disk address in addition to the starting (or low) disk address.
 Size (Default) Show size occupied in addition to the starting (or low) disk address.

Expire: controls the inclusion of expired entries (where the entry expiration date is not a future date) in VTOC display (only the first letter of each choice need be keyed):

- Yes (Default) Include expired entries (which are flagged with an asterisk between the creation and expiration dates).
 No Exclude expired entries and show that space as available space (indicated as 'Free.Space')

All remaining screen lines beginning with screen line 3 are divided into 7 columns and will contain a list of all DASD entries available for display, filtered by any specification other than NONE at the POOL field on screen line 1.

You can key the cuu of your choice in the first screen line just to the right of the constant 'CUU' or position the cursor anywhere on one of the list of available DASD's (shown as cuu=volser in up to 7 columns beginning on screen line 3) and pressing enter. To refresh the list of available DASD's displayed to be based on a new poolname, enter the desired poolname to the right of the constant 'POOL', clear any value found to the right of the constant 'CUU', and press *enter*.

VTOC Screen

The VTOC Screen is displayed when the TVTO transaction-id and either a cuu or volser is entered from a blank CICS screen or from a successful selection made from the HELP screen.

Screen line 1 includes the cuu, volser, and DASD model for the unit being displayed as well as the current date, time, current and last screen page numbers.

Screen line 2 provides titles for the columns that occupy all the remaining screen lines (except the last) beginning on screen line 3:

File Name: The full 44-character name of the file occupying the this extent (may also be shown right justified as 'Free.Space' indicating available space, 'Volume.Table.Of.Contents' indicating that this space is occupied by the VTOC, 'Begin pool poolname' indicating the lower limit for this poolname, or 'End pool poolname' indicating the upper limit for this poolname).

Low: Starting disk address for this entry.

High: Ending disk address for this entry.

Size: Amount of disk space occupied by this entry.

Seq: The sequence number of this extent on all volumes occupied by this File Name.

Created: The julian date (in the format yyyyddd) when this entry was placed in the VTOC.

Expires: The julian date (in the format yyyyddd) when this entry is scheduled to expire (if this date is not greater then the corresponding 'Created' date, an asterisk (*) is placed between this date and the corresponding 'Created' date).

The last screen line shows the action taken for each of the programmed PF keys and also the total number of free (available) tracks or blocks in the total number of free space extents.

BIM-EPIC VTOC Display for 140=DOSRES (9336-10)					1997/345	15:31:52	Page	1 of	1
File Name					Low	Size	Seq	Created	Expires
VSE.SYSRES.LIBRARY					2	59134	000	1996337	1999365
DOS.LABEL.FILE.FF0000017490.AREA1					59136	320	000	1997255	1999366
VSE.POWER.QUEUE.FILE					59456	384	000	1997013	1999366
Free.Space					59840	384			
Z9999996.VSAMDSPC.TADDB87D.T1DF884A					60224	2880	000	1996337	1999366
Z9999992.VSAMDSPC.TADDB87D.T90FF048					63104	62400	000	1996337	1999366
Volume.Table.Of.Contents					125504	64			
DOS.PAGING.FILE.FF0000017490					125568	16896	000	1997273	1999366
Z9999992.VSAMDSPC.TADDB87D.T3C13E4E					142464	155520	000	1996337	1999366
Free.Space					297984	12032			
VSE.SYSTEM.HISTORY.FILE					310016	5056	000	1997345	1999366
DOS.PAGING.FILE.FF0000017490					315072	16384	002	1997273	1999366
DOS.PAGING.FILE.FF0000017490					331456	319488	003	1997273	1999366
Free.Space					650944	270578			
PF 1=HLP 2=RFSH 3=END 4=FST 5=LST 7=BWD 8=FWD					282994	Free Blocks in	3	Extents	

Valid PF keys

PF Key	Function	Explanation
PF1	HELP	Display help screen.
PF2	RFSH	Read all VTOC entries again.
PF3	END	Return to CICS.
PF4	FST	Go to the first screen.
PF5	LST	Go to the last screen.
PF7	BWD	Back one screen.
PF8	FWD	Forward one screen.

Offload System

Introduction

Description

BIM-EPIC provides offload/onload, backup/restore, offload version consolidation, and offload version copy functions for controlled sequential disk dataset versions; and is only available on BIM-EPIC systems supporting both disk and tape management. The dataset population to be processed by the offload, backup, version consolidation, and version copy functions are identified by control statements created either manually or by the offload system itself.

The offload/onload function only processes controlled sequential disk dataset versions identified in the BIM-EPIC catalog. The offload function copies a dataset version to tape and then deletes the disk VTOC entry leaving the version identified as residing on tape only. The onload function copies a dataset version identified as residing on tape only back to disk identifying the version as residing on disk only. Onloading a version already identified as residing on disk only will function the same as the restore function.

The backup/restore function processes controlled sequential disk dataset versions identified in the BIM-EPIC catalog as well as controlled start-track-1 sequential disk datasets. The backup function copies a dataset version to tape leaving the disk version in place. The restore function copies a dataset to disk creating a new version (for generation files) or replacing the current version (for work files).

When copying a dataset to disk, the dataset characteristics currently in effect reflecting changes in DASD architecture and geometry (i.e., CKD to FBA, 3380 to 3390, etc.) will be honored. Disk leveling is also honored when writing multiple datasets as well as multiple extents for a single dataset.

The Offload Version Consolidation function consolidates one or more offload versions into a new version. This can be useful for offload versions that have most of their datasets in scratch status. As an example: An Offload Dataset has Ver 5 with 4 tapes (80% of its files in scratch status) and Ver 7 with 3 tapes (70% in scratch status). These two versions could be consolidated into a new version of perhaps 1 or 2 tapes, - freeing 5 tapes for other uses.

The Offload Version Copy function copies an offload version into BIM-EPIC backup format. It will copy an existing backup version or an existing offload version.

Programs

The offload system consists of four programs:

Program	Function
TSIDOFD	Selects datasets for offload and backup by creating control statements for TSIDOFD.
TSIDOFB	Offloads and backs up selected datasets.
TSIDONL	Restores offload and backup datasets back to disk and creates a listing of datasets on the offload tape.
TSIDOFD	Consolidates offload tapes and copies offload and backup tapes.

High Performance I/O

The BIM-EPIC offload system uses proprietary disk and tape I/O routines that provide extraordinary high performance through the use of physical I/O, maximum block sizes, and a single tape file. The offload system is implemented automatically as soon as BIM-EPIC is activated and is fully upward compatible with the prior offload system (all JCL, control statements, program names, and procedures remain unchanged). This onload program automatically recognizes and processes offload and backup tapes created under the prior (BIM-EPIC 4.14 and prior) offload system so complete access to existing offload and backup datasets is assured.

IMPORTANT

Whenever an offloaded dataset is read, this high performance feature requires sufficient disk space to temporarily copy the data to the disk pool specified in the EDD. If an “insufficient space” condition occurs while creating this temporary disk file, BIM-EPIC allocation configuration options (e.g., FRAG and alternate pools) will be used.

Selecting Datasets To Be Offloaded (TSIDOFD)

Description

TSIDOFD creates control statements for input to TSIDOFD based on selection criteria you specify. These control statements determine which datasets will be offloaded or backed up. You specify selection criteria to TSIDOFD using INCLUDE and OMIT control statements.

Selection Criteria for Disk Dataset Versions Identified in the BIM-EPIC Catalog

Datasets can be included for offload/backup by:

- pool name
- volume serial number
- dataset name
- dataset names beginning with a specific character string
- version number
- versions created since the last time the TSIDUTL program was executed with the DSPLY NEW function
- user ID
- system ID
- access within the last n days (SLA)
- no access within the last n days (DLA)

SLA and DLA are mutually exclusive.

Datasets can be omitted from the inclusion list by:

- pool name
 - volume serial number
 - dataset name
 - dataset names beginning with a specific character string
 - version number
 - user ID
 - system ID
-

Selection Criteria for Start-Track-1 Disk Datasets

Datasets can be included for backup by:

- pool name
- dataset name
- dataset names beginning with a specific character string

Datasets can be omitted from the inclusion list by:

- dataset name
- dataset names beginning with a specific character string

JCL Requirements

```

[// xLBL OUTPUT,'sysin.data.set']
//EXEC TSIDOFD
control statements
/*

```

An internal table is used during the dataset selection process to store each version selected. The capacity of this table is dependent on the amount of excess partition storage available (up to and not including partition GETVIS storage) and will never be less than 100. This internal table size is increased by approximately 21 entries for each 1K of excess partition storage available beyond the first 5K. For example, if SIZE=(TSIDOFD,50K) is specified on the EXEC statement, then the internal table capacity would be about 1066 entries. Specifying a SIZE operand value that results in about 200K for partition GETVIS maximizes this internal table. For example, when utilizing a 2048K partition (including GETVIS), a SIZE operand value of 1848K would be appropriate.

If the DATASET SELECTION REPORT is produced, the actual capacity of the internal table for the current processing is stated on the second report heading line. If this program attempts to select more dataset versions than can be held in this internal table, no output will be created and this program will terminate with a return code of 4 after issuing message EP618.

UPSI Values

None

Dataset Requirements

The control statements created by TSIDOFD are created on SYSPCH if SYS005 is assigned to SYSPCH. Otherwise, control statements can be written to tape or disk by supplying a TLBL, DLBL, or DD with the DTF name OUTPUT.

Control Statements

Dataset selection criteria are specified using INCLUDE and OMIT control statements. You can specify multiple parameters on any INCLUDE or OMIT statement, but parameter expressions cannot be continued. Multiple INCLUDE and OMIT statements can be submitted reflecting the DSN=, POL=, and VOL= parameters up to the following limits:

- 25 dataset names (DSN=) for INCLUDE processing
- 25 dataset names (DSN=) for OMIT processing
- 50 pool names (POL=) for INCLUDE processing
- 50 pool names (POL=) for OMIT processing
- 50 volume serial numbers (VOL=) for include processing
- 50 volume serial numbers (VOL=) for omit processing

An internal table is used during the dataset selection process to store each version selected. The capacity of this table is dependent on the amount of excess partition storage available up to and not including partition GETVIS storage, but will never be less than 100. If the DATASET SELECTION REPORT is produced, the actual capacity of the internal table for the current processing is stated on the second report heading line.

AND logic always applies when multiple selection criteria is specified, regardless of whether submitted on a single control statement, multiple control statements, or both.

One primary set of parentheses must enclose all the parameter expressions on a single control statement. Commas or blanks can separate multiple parameter expressions. The parameters ALLVER, VOL, and POL each allow you to specify more than one value; in such cases, the values must be separated by commas within a set of parentheses.

Syntax

REPORT=YES | NO

CATALOG=YES | NO

```

INCLUDE  ( { [ DSN='dataset-name' ]
              [ ,VER=n ]
              [ DSN='string.ALL' ]
              [ ALLVER=n ]
              [ DLA=n ]
              [ UID=x ]
              [ SID=x ]
              [ SLA=n ]
              [ POL=poolid,poolid,... ]
              [ VOL=volid,volid,... ]
              [ NEW ]
            } )

OMIT     ( { [ DSN='dataset-name' ]
              [ ,VER=n ]
              [ DSN='string.all' ]
              [ ALLVER=n ]
              [ UID=x ]
              [ SID=x ]
              [ POL=poolid,poolid,... ]
              [ VOL=volid,volid,... ]
            } )

```

INCLUDE/OMIT Parameters

Parameter	Required With CATALOG=YES?	Required With CATALOG=NO?	Valid	Default
ALLVER	No	Not Allowed	1 to 9999	None
DLA	No	Not Allowed	1 to 365	None
DSN	No	No	1 to 44 characters	None
NEW	No	Not Allowed	None	OFF
POL	No	Yes	1 to 6 characters	None
SID	No	Not Allowed	2 characters	None
SLA	No	Not Allowed	1 to 365	None
UID	No	Not Allowed	2 characters	None
VER	No	Not Allowed	1 to 9999	None
VOL	No	Not Allowed	1 to 6 characters	None

REPORT specifies whether the Preliminary Dataset Selection Report is to be printed and defaults to YES. This report shows the size of the internal selection table (in heading 2), the selection criteria used, and the actual dataset versions selected.

CATALOG specifies whether selection should be for controlled disk datasets found in the BIM-EPIC catalog or for controlled start-track-1 disk datasets. CATALOG=YES indicates selection should be for controlled disk datasets found in the BIM-EPIC catalog and is the default. CATALOG=NO indicates selection should be for controlled start-track-1 disk datasets.

INCLUDE specifies that the datasets to be included for offload or backup are identified by the subparameters that follow. At least one INCLUDE statement must be used.

OMIT specifies that the datasets to be omitted from offload or backup are identified by the subparameters that follow. OMIT acts on datasets that have been included with an INCLUDE statement.

POL selects datasets that are allocated to the specified pool. Up to 10 pool IDs can be specified with INCLUDE, and for CATALOG=YES only, another 10 can be specified for OMIT. You can specify multiple pool IDs on a single POL parameter (POL=pool01,pool02,pool03...)

For CATALOG=YES, the POL parameter is checked against only the *current* pool name in the dataset's definition. If that pool name has changed after a dataset version was created, you must use other options to include or omit that dataset version.

For CATALOG=NO, the dataset must have been created using the specified pool name and also satisfy all the following additional criteria:

- SAM file
- Prime Data extents
- Reside completely within the *current* pool limits
- *NOT* be defined in the BIM-EPIC catalog

Additionally for CATALOG=NO, datasets created by PIOCS programs (e.g., programs that do not use Logical IOCS and also do not have an interface with BIM-EPIC) will *NOT* be selected unless specifically requested using an INCLUDE control statement with a fully qualified DSN parameter (i.e., without '.ALL').

VOL selects datasets from the specified disk volume serial number. Up to 10 volume serial numbers can be specified for INCLUDE, and another 10 can be specified for OMIT. You can specify multiple volume serial numbers on a single VOL parameter (VOL=vol001, vol002, vol003,...)

DSN selects a single dataset name or a set of datasets beginning with the same character string. The value specified must be enclosed in apostrophes. To specify a group of datasets, specify '.ALL' at the end of the dataset name.

VER selects a version number for a particular dataset. It can only be supplied if the DSN parameter is also supplied. If omitted, all versions of the dataset are selected.

The following subparameters can be specified only once for INCLUDE and once for OMIT.

ALLVER selects datasets by version number. ALLVER affects all datasets to be selected (VER affects only a specific dataset). You can specify up to five versions with a single ALLVER parameter. (e.g., ALLVER=n,n,n,n,n).

SID selects datasets with the specified system ID.

UID selects datasets with the specified user ID.

DLA selects datasets which have not been accessed for the specified number of days. DLA is valid only for INCLUDE.

SLA selects datasets which have been accessed within the specified number of days. SLA is valid only for INCLUDE.

NEW selects datasets which have been created since the last time the TSIDUTL program was executed with the DSPLY NEW function. NEW is valid only for INCLUDE.

Offloading and Backing Up Datasets (TSIDOFI)

Description

TSIDOFI offloads or backs up selected datasets according to the settings on an UPSI statement and the contents of control statements.

Control statements indicating which datasets are to be processed can be generated by TSIDOFI or specified manually. They are supplied as SYSIPT data. TSIDOFI sorts the control statements and copies the datasets to tape in order by descending dataset name and version number.

The offload tape is always associated with a cataloged dataset. You can use the default offload dataset OFFLOAD.DATASET, or you can create your own dataset. If you create your own dataset, you must catalog it and supply a TLBL or DD statement using the DTF name OUTPUT.

A tape version containing offloaded disk versions is not eligible for scratch until all the offloaded disk versions either expire or are unloaded back to disk and then not until any additional retention criteria attributed to the tape version has also been satisfied.

JCL Requirements

When the control statements are supplied as SYSIPT data:

```
[// TLBL OUTPUT,'offload.dataset.name']
// UPSI xx
// EXEC TSIDOFI
[control statements]
/*
```

When the control statements are supplied from a tape or disk dataset (using GDI for SYSIPT):

```
[// TLBL OUTPUT,'offload.dataset.name']
// UPSI xx
* //IJSYSIN DD DSN=sysin.dataset,DISP=(OLD,DELETE,KEEP)
// EXEC TSIDOFI
/*
```

When the control statements are supplied from a tape or disk dataset: (using a standard VSE assignment for SYSIPT):

```
{// TLBL OUTPUT,'offload.dataset.name'}
// DLBL IJSYSIN,'sysin.dataset'
// EXTENT SYSIPT
ASSGN SYSIPT,DISK,VOL=volser,SHR
// UPSI xx
// EXEC TSIDOFI
/*
CLOSE SYSIPT,SYSRDR

{// TLBL OUTPUT,'offload.dataset.name'}
// TLBL IJSYSIN,'sysin.dataset'
// EXEC TSIDASS
OPEN IJSYSIN,SYS020,INPUT
/*
// ASSGN SYSIPT,SYS020
// UPSI xx
// EXEC TSIDOFI
```

```
/*
// RESET SYSIPT
```

UPSI Values

UPSI 0 specifies OFFLOAD.

UPSI 1 specifies BACKUP. Note that if control statements for start-track-1 datasets are submitted, UPSI 1 is automatically forced.

UPSI x1 specifies special handling when the number of control statements exceeds the internal work area used by TSIDOFD. When this UPSI setting is used, the control statements are not sorted, and datasets are offloaded in the order in which the control statements are submitted.

UPSI xx1 specifies that DSN selection will be by Generation Number rather than by Version Number. DO NOT USE THIS UPSI if selection statements were created by the TSIDOFD program.

WARNING!

If UPSI x1 is used with an execution of TSIDOFD where the TSIDOFD control statements were NOT created by the TSIDOFD program, it becomes the user's responsibility to assure that, when multiple versions of a dataset are restored using TSIDONL, the oldest version is restored first, the second oldest version is restored second, and so on.

Dataset Requirements

OUTPUT specifies a tape dataset to control the offload tape. *This dataset must be cataloged.* If a label is not supplied for the OUTPUT dataset, TSIDOFD catalogs and uses the dataset name OFFLOAD.DATA.SET.

IJSYSIN specifies that the control statements reside on a tape or disk dataset that was created by the TSIDOFD program. You can use either standard VSE assignments or GDI.

PARM Values

PARM='DLAUPD=NO' can be specified on the EXEC Job Control statement or command to prevent the current job name and date from being stored as the last accessed information for each version read by the TSIDOFD program. By specifying PARM='DLAUPD=YES' or not specifying the PARM operand, the current job name and date will be stored as the last accessed information for each version read by the TSIDOFD program.

Control Statements

Use control statements to indicate which datasets to offload or backup. These can be generated with TSIDOFD. Control statements for disk datasets versions identified in the BIM-EPIC catalog and control statements for start-track-1 datasets *cannot* be mixed together for input to the same execution of TSIDOFD.

If UPSI xx1 is used, control statements must not have been generated by the TSIDOFD program.

Syntax for Sequential Disk Versions Identified in the BIM-EPIC Catalog

'dataset-name',version-number
'dataset-name',generation-number

Syntax for Start-Track-1 Disk Datasets

'dataset-name',POL=poolnm

Parameters

Parameter	Required?	Valid	Default
<i>dataset-name</i>	Yes	1 to 44 characters	None
<i>version/generation number</i>	No	1 to 9999/1-65534	1/None
<i>poolnm</i>	Yes	6 characters	None

dataset-name specifies the name of the dataset to be offloaded.

version specifies the version to be offloaded.

generation specifies the generation number to be offloaded. UPSI xx1 must be set, and the input control statements must not have been created by TSIDOFD.

"*version/generation number*" is positional and only one number can be specified. If UPSI xx1 is not set, EPIC regards the number as a version number. If UPSI xx1 is set, the number is regarded as a generation-number.

Poolnm specifies the name of the pool where the dataset resides.

Restart Information

TSIDOFD copies each dataset that is to be backed up or offloaded to the output tape. In addition, the OFFLOAD function deletes the VTOC entry for each dataset as soon as the copy function has completed. If for any reason a backup or offload job that involves more than one dataset terminates abnormally and must be restarted, please note the following special restart considerations.

Restarting Backups

Backups must be restarted from the beginning because the control statements are sorted internally by TSIDOFD.

Step	Action
1	Scratch the incomplete backup dataset version.
2	Restart the job in the TSIDOFD step.

Restarting Offloads

The same dataset cannot be offloaded twice, so TSIDOFI automatically restarts with the next dataset to be offloaded. The datasets already offloaded are in the incomplete offload dataset version.

Step	Action
1	Do not scratch the incomplete offload dataset version.
2	Restart the job in the TSIDOFI step. Disregard any error messages for missing datasets. (These were probably offloaded by the run which did not complete.)
3	Keep the TSIDOFI offload report and tapes from both the incomplete and restarted runs.
4	If the offloaded datasets are subsequently unloaded, run TSIDONL with the offload dataset versions from both the incomplete and restarted runs.

Examples

The following job offloads datasets based on the control statements supplied using the SYSIPT dataset SYSIN.DATASET. This input dataset is deleted when it is closed. Because no TLBL or DD for OUTPUT was supplied, all datasets will be offload to a tape version that is retained under the output dataset name OFFLOAD.DATASET.

```
// JOB OFFLOAD
* //IJSYSIN DD DSN=SYSIN.DATASET,DISP=(OLD,DELETE)
// UPSI 0
// EXEC TSIDOFI
/*
/ &
```

The following job backs up datasets based on the control statements supplied using the SYSIPT dataset SYSIN.DATASET. This input dataset is deleted when it is closed. Because a TLBL for OUTPUT was supplied, all datasets will be backed up to a tape version that is retained under the output dataset name BACKUP.DATASET.

```
// JOB BACKUP
* //IJSYSIN DD DSN=SYSIN.DATASET,DISP=(OLD,DELETE)
// TLBL OUTPUT, 'BACKUP.DATASET'
// UPSI 1
// EXEC TSIDOFI
/*
/ &
```

The following job offloads four datasets named on the control statements supplied as SYSIPT data. All datasets are offloaded to a tape version that is retained under the output dataset name "USER.OFFLOAD.DATASET".

```
// JOB OFFLOAD
// TLBL OUTPUT, 'USER.OFFLOAD.DATASET'
// UPSI 0
// EXEC TSIDOFI
'PAYROLL.CHECKS',2
'BILLING.MASTER',9
'BILLING.MASTER',10
'RECON.TRANS'
/*
/ &
```

The following job backs up five start-track-1 datasets named on the control statements supplied as SYSIPT data. All datasets are offloaded to a tape version that is retained under the output dataset name "START.TRACK1.DATA".

```
// JOB BACKUP
// TLBL OUTPUT, 'START.TRACK1.DATA'
// UPSI 1
// EXEC TSIDOFI
`SNOOPY.DOG',POL=WDPOOL
`MICKEY.MOUSE',POL=WDPOOL
`TAZ.DEVIL',POL=WBPOOL
`WILEY.COYOTEIUS.MAXIMUMUS.HURTUM',POL=WBPOOL
`CRAYZY.RABBUTTE',POL=WBPOOL
/*
/&
```

Onloading and Restoring Datasets (TSIDONL)

Description

TSIDONL onloads or restores datasets back to disk. It can also be used to produce a report listing the datasets contained on the offload tape. TSIDONL functions are selected using UPSI values.

You can onload a dataset from an offload tape only. TSIDONL assigns the dataset a current relative version based upon its absolute generation number. You can restore a dataset from either an offload or a backup tape. For sequential disk versions identified in the BIM-EPIC catalog, the restore function assigns the dataset version number 1.

Configuration Options

The Start Track 1 feature is required to automatically access a dataset that has been offloaded. Specify a starting track or block address for this feature using the STRTRK option.

JCL Requirements

```
// UPSI xxxx
[// TLBL INPUT, 'user-dataset-name' ]
// EXEC TSIDONL
[control statements]
/*
```

UPSI Values

Value	Function
000	Onloads all datasets from the offload tape. No control statements are read.
010	Onloads datasets selected by control statements only.
100	Restores all datasets from the offload or backup tape. No control statements are read.
110	Restores datasets selected by control statements only.
xx1	Creates the report only. No datasets are onloaded or restored. No control statements are read.
xxx1	Assumes uncataloged input tapes are in new offload format. (See note below).
0x0x1	Prevents onloading of a dataset that has been scratched from the BIM-EPIC Catalog. (See note below).

UPSI xxx1

UPSI xxx1, which only applies to tapes containing offload or backup data for sequential disk versions identified in the BIM-EPIC catalog, should *only* be used when the tape volume(s) accessed are not cataloged *AND* they were created using the new proprietary high performance offload system (BIM-EPIC 4.15 and higher). The following events could cause this situation:

- The tape volume was PURGED from the catalog

- The tape volume was created at a different site and is not known to the catalog at the site where it is being restored

In all cases, the disk dataset to which a tape is being restored must be defined to the BIM-EPIC catalog. Instead of using UPSI xxx1 for an uncataloged tape, we recommend that you ADD the tape volume to the BIM-EPIC catalog and then RETAIN it using the OF2 parameter.

UPSI 0x0x1

BIM-EPIC defaults to restoring a dataset as version 1 when attempting to onload a dataset that has been scratched from the BIM-EPIC catalog. By setting UPSI bit 4 on, BIM-EPIC will not onload any dataset found on the input tape that has been scratched from the BIM-EPIC Catalog.

UPSI bit 4 is only effective when doing an onload function (UPSI bit 0 off) and *not* a report-only processing (UPSI bit 2 on). If UPSI bit 4 is off and a dataset is restored as version 1 because it has been scratched from the BIM-EPIC Catalog, the characters “SCR'D” will appear on the TSIDONL report entry for that dataset version between the dataset name and its old generation number.

Dataset Requirements

INPUT specifies the offload dataset name used when TSIDOFI was executed. If no TLBL or DD statement is supplied, TSIDONL uses version 1 of the dataset 'OFFLOAD.DATASET'. No statements are required for the output datasets.

TSIDONL Control Statements

Description

Control statements are read if UPSI 010 or 110 is specified. There are two types of control statements for TSIDONL:

- A *dataset-name* control statement specifies a dataset to be onloaded or restored.
- A START control statement begins an onload or restore at a particular dataset on the tape.

Continuation is possible by ending with the comma following a parameter and starting the next parameter on the next control statement.

If no control statements are supplied when UPSI x10 is specified, all datasets on the tape are written to disk.

Syntax for Sequential Disk Versions Identified in the BIM-EPIC Catalog

'dataset-name', generation-number

Syntax for Start-Track-1 Disk Datasets

'dataset-name', POL=poolnm, 'alt-dataset-name'

Parameters

Parameters	Required?	Valid	Default
<i>dataset-name</i>	Yes	1 to 44 characters	None
<i>generation-number</i>	see below	1 to 999999	None
<i>Poolnm</i>	No	6 characters	see below
<i>alt-dataset-name</i>	No	1-44 characters	None

dataset-name specifies the name of the dataset to be onloaded or restored.

generation-number specifies the absolute generation number of the tape dataset to be onloaded or restored. To specify all generations of a dataset, code an asterisk (*). The generation number can be omitted for a work dataset.

Poolnm specifies the name of the pool where the dataset is to be restored. If not specified, the dataset will be restored to the same pool where the dataset originally resided.

Alt-dataset-name is the dataset-name to be used for the restored dataset. If not specified, the dataset-name used will be *dataset-name*.

Restart Information

Each backed up or offloaded dataset is processed as a separate entity. Therefore, if for any reason a multi-dataset onload or restore job terminates abnormally and is restarted, special restart considerations are necessary. In order to avoid dataset duplication and maintain version order integrity, the job must be restarted on the dataset being processed at the time the abend occurred. Use the following procedure.

Step	Action
1	Refer to the restore/onload report and obtain the last dataset name and generation listed.
2	Compare this to the backup/offload report and obtain the next dataset name (and generation number, if appropriate).
3	Restart TSIDONL using a START control statement specifying the dataset name and generation from Step 2.

Syntax

START= '*dataset-name*' , *generation-number*

dataset-name specifies the name of the dataset with which to begin onload or restore processing.

generation-number specifies the absolute generation number of the disk dataset. The generation number can be omitted for a work dataset identified in the BIM-EPIC catalog and *MUST* be omitted for a start-track-1 dataset.

Examples

The following job onloads all the datasets on the tape. It uses version 1 of the default offload dataset. The datasets remain assigned to their current relative version number (based upon absolute generation number) unless that version has been scratched. If the version has been scratched, they are assigned to version number 1.

```
// JOB ONLOAD
// EXEC TSIDONL
/*
/ &
```

The following job restores all generations of the ACCOUNTS.PAY dataset. Each generation is assigned version number 1 as it is restored. The last version restored is version number 1. Because datasets are copied in reverse version order, the last version restored is the newest version on the tape.

```
// JOB RESTORE
// UPSI 110
// EXEC TSIDONL
'ACCOUNTS.PAY' , *
/*
/ &
```

The following job onloads all datasets starting with and written after generation 112 of the ACCOUNTS.PAY dataset. The datasets remain assigned to their current relative version number unless that version has been scratched. If the version has been scratched, they are assigned to version number 1. The job uses version 1 of the USER.DATA.SET offload tape.

```
// JOB ONLOAD
// TLBL INPUT, 'USER.DATA.SET'
// UPSI 010
// EXEC TSIDONL
START= 'ACCOUNTS.PAY' ,112
/*
/ &
```

The following job restores all datasets found on the start-track-1 disk dataset backup tape.

```
// JOB RESTORE
// UPSI 100
// EXEC TSIDONL
/*
/ &
```

The following job restores the three start-track-1 datasets named on the control statements supplied as SYSIPT data. The second control statements reflects the POL= parameter to cause that start-track-1 dataset only to be restored to the specified pool (WDPOOL) rather than the pool where the dataset originally resided. The fourth control statement reflects both the POL= parameter and an alternate dataset name to cause that start-track-1 dataset only to be restored to the specified pool (DPOUND) using the alternate dataset name ('WOODSTOCK') rather than the pool and dataset name of the original file. The last control statement (which continues across two records) reflects an alternate dataset name to cause that start-track-1 dataset only to be restored using the alternate dataset name ('SYLVESTER') rather than the dataset name of the original file.

```
// JOB RESTORE
// UPSI 110
// EXEC TSIDONL
'WILEY.COYOTEIUS.MAXIMUMUS.HURTUM.IV.CERTUM'
'CRAYZY.WABBUTTE' ,POL=WDPOOL
'TAZ.DEVIL'
'SNOOPY' ,POL=DPOUND, 'WOODSTOCK'
'TWEETY' ,
'SYLVESTER'
/*
/ &
```

Differences Between TSIDONL Onload and TSIDONL Restore

Description

Suppose we have the following versions of a dataset called DATASET.A:

Version number	Absolute generation number (AGN)
1	46
2	45
3	44
4	43
5	42
6	41

Absolute generation numbers (AGN) run in inverse sequence to the version numbers within a dataset. At one time (when it was first created), version 2, AGN=45 was version 1. The 45 means that this was the 45th time this dataset was created. The next time a version was created, it was the 46th, hence the AGN=46. The newly created version became version 1, the most current version.

Suppose versions 3 through 6 were offloaded to tape. Now DATASET.A looks as follows:

Version number	Absolute generation number (AGN)
1	46
2	45

Version information still exists in DATASET.A for versions 3 through 6, but now those versions display as tape datasets instead of disk. Versions 3 through 6 are still available for scratching (assuming all retention criteria has been met) even though they have been offloaded.

Suppose versions 3 through 6 are not scratched and two new versions are created. Now DATASET.A looks as follows:

Version number	Absolute generation number (AGN)
1	48
2	47
3	46
4	45

Suppose now we onload our previously offloaded datasets. After doing this, DATASET.A would look as follows:

Version number	Absolute generation number (AGN)
1	48
2	47
3	46
4	45
5 (old ver. 3)	52 (AGN is used to find where the onloaded datasets belong)
6 (old ver. 4)	51
7 (old ver.5)	50
8 (old ver.6)	49

The offloaded datasets were onloaded according to their AGNs, so they are no longer version 3 through 6, but versions 5 through 8.

Now suppose we restore our previously offloaded datasets. After doing this, DATASET.A would look as follows:

'DATASET.A'

Version number	Absolute generation number (AGN)
1 (old ver.3)	52 (old AGN=44)
2 (old ver.4)	51 (old AGN=43)
3 (old ver.5)	50 (old AGN=42)
4 (old ver.6)	49 (old AGN=41)
5	48
6	47
7	46
8	45

Each restored version is restored as version 1. AGN=41 is restored first, AGN=42 second, and so on. As each dataset is restored, it receives a new AGN.

Consolidating Offload Versions (TSIDOFB)

Description

TSIDOFB consolidates one or more offload versions into a single version.. Control cards are used to report on consolidation candidates and to specify the versions to be consolidated.

JCL Requirements

Control statements are supplied as SYSIPT data:

```
// TLBL OFCINP,'inputdatasetname'
// TLBL OFCOUT,'outputdatasetname'
// DLBL SORTWK1,'datasetname'
// EXEC TSIDOFB
control statements
/*
```

TLBL's are not required for the REPT function

DLBL for SORTWK1 is required for REPT function only, and can be omitted if it is in Standard Labels.

UPSI Values

None

Dataset Requirements

The dataset names used in OFCINP and OFCOUT must be the same if version consolidation (CONI) is being done. They must be different if COPY is being done.

Offload and Backup data must be in FAVER2 format. All offloads and backups created in EPIC 4.2 and above satisfy this criteria.

Control Statements

These control statements are used:

Statement	Purpose
REPT	Produce an <i>Offload Consolidation Candidate Report</i> . The report provides a list of all offload datasets in the BIM-EPIC catalog and their environmental detail (media type, expire date, volume serial number, ...). If requested, appropriate datasets will be flagged as "consolidation candidates".
CONI	Specify the versions that are to be consolidated into a new version.
CONO	Specify the version number of the consolidated output. If not present, the new version will be 1.
COPY	Specify the offload or backup version that is to be copied into a new version.

Syntax

Control statements can begin in any column. Continuations cannot be used.

REPT [PCT=*n*]

If used, it must be the only control statement submitted.

PCT=*n* is optional and is meaningful only for offload datasets that reside on cartridge media. If present, the *Offload Consolidation Report* will print *Consolidation Candidate* next to each dataset version whose percentage of “number of blocks in non-scratch status” is equal to or less than this number. An example: If PCT=30 is specified and version 7 of an offload dataset has 25% of its available blocks in use, it will have *Consolidation Candidate* listed next to its report line. “Available blocks” is based on an estimate of the number of 64K blocks that will fit on the various cartridge media. “Number of available blocks in non-scratch status” is taken from data in the BIM-EPIC catalog.

Examples: REPT
REPT PCT=40

CONI *n* *n* *n* ... *n*

n is a version number that is input to the consolidation process. There must be at least one value present. Each value is separated by a space. There can be as many versions specified as will fit on an 80 column SYSIPT input.

Only one CONI statement can be present.

Unless the CONO statement is used, the version number of the consolidated output will be 1.

Upon completion of the consolidation, the input versions are set to scratch status in the BIM-EPIC catalog.

Examples: CONI 5 12
CONI 6 14 23 30

CONO NV=*n*

n is the version number to be assigned to the consolidated output version if the default of 1 is not desired. The assignment is not made until the consolidation has completed and the input versions have been scratched (see CONI, above). At that time, if the NV number would cause a “sequence gap”, the version will remain at 1. To clarify:

At the beginning, versions 1 2 3 4 5 exist. Control statements are:

```
CONI 2 3
CONO NV=4
```

At completion of consolidation and prior to CONO processing, versions are

- 1 – result of the consolidation
 - 2 – previously version 4
 - 3 – previously version 5
- The original versions 2 and 3 are scratched.

CONO's NV=4 would cause a sequence gap since there is no version 3. NV=3 or 2 would be ok since no sequence gap would occur.

CONO must come after CONI.

Example: CONI 5 12
 CONO NV=5

COPY n

n is the version number of the dataset that is input to the copy process. The version number of the output dataset will be 1. Only one version can be copied at a time. The input and output dataset names must be different.

Only one COPY statement can be present.

Example: COPY 4

Examples

The following job produces an *Offload Consolidation Candidate Report* .

```
// JOB OFCREP
// DLBL SORTWK1, 'SORT.WK1 '
// EXEC TSIDOFD
REPT
/*
/ &
```

The following job produces an *Offload Consolidation Candidate Report* and identifies versions that are candidates for consolidation.

```
// JOB OFCREP
// DLBL SORTWK1, 'SORT.WK1 '
// EXEC TSIDOFD
REPT PCT=35
/*
/ &
```

The following job consolidates versions 4, 8, 11 into a new version 1.

```
// JOB OFCCONS
// TLBL OFCINP, 'PAYR.MONTHLY.OFLD'
// TLBL OFCOUT, 'PAYR.MONTHLY.OFLD'
// EXEC TSIDOFD
CONI 4 8 11
/*
/ &
```

The following job consolidates versions 4, 8, 11 into a new version 6.

```
// JOB OFCCONS
// TLBL OFCINP, 'PAYR.MONTHLY.OFLD'
// TLBL OFCOUT, 'PAYR.MONTHLY.OFLD'
// EXEC TSIDOFD
CONI 4 8 11
CONO NV=6
/*
/ &
```

The following job copies offload version 7 and creates version 1 in PAYR.MON.COPY. The new version 1 will be in BACKUP format. Version 7 may be in OFFLOAD or BACKUP format.

```
// JOB OFCCONS
// TLBL OFCINP,'PAYR.MONTHLY.OFLD'
// TLBL OFCOUT,'PAYR.MON.COPY'
// EXEC TSIDOFD
COPY 7
/*
/ &
```


A

BIM-EPIC Report Fields and Examples

Standard reports are listed by program name. Programs are presented in alphabetical order.

Chapter Contents

Customized Reports.....	A-2
TOJA Report Examples.....	A-3
TSIDARP Report.....	A-6
TSIDBRP Reports	A-8
TSIDDIS Report	A-11
TSIDJAC Report	A-13
TSIDLVT Report.....	A-15
TSIDMNT Reports.....	A-17
TSIDOFD Report	A-19
TSIDOFL Report.....	A-20
TSIDONL Report	A-21
TSIDPRV Report.....	A-22
TSIDSDS Report	A-23
TSIDTAC Report	A-24
TSIDUTL Reports	A-26
Volume Serial Number and Vault Sequences	A-29
Display Vault Report	A-32
Display Scratch Report	A-33
Projected Scratch (PSCRATCH) Report	A-34
TSIDVLT Report.....	A-35

Customized Reports

Description

In addition to the standard reports, you can use your own report writers to produce customized reports from BIM-EPIC data.

The DSN Catalog contains all tape and dataset information, including BIM-EPIC dataset definitions (EDDs) and version data. We have provided the TSIDAC2 subroutine for access to the DSN Catalog. See Chapter 4 for procedures.

The Recorder File contains all DSN Catalog changes, job accounting information, and user comments. You can use the Recorder to report data such as catalog use and machine use. Use the Recorder copybook TSIDREC.C to generate reports from this the Recorder File.

TOJA Report Examples

Description

TOJA produces job accounting information online. For information on producing TOJA reports, see "Online Access to the Job Accounting Database" in Chapter 11.

The report has three distinct levels: the Job Overview panel, the Program Overview panel, and the Program Audit panel. Each displays job accounting statistics in increasing detail.

Job Overview Panel

The Job Overview panel presents an overview of execution based on a jobname. For each job, there is a one line entry for each program executed in that job.

BIM-EPIC/VSE ONLINE JOB ACCOUNTING FACILITY								
.ALL								PAGE 02
JOB NAME	PHASE NAME	START DATE	START TIME	DURATION	CC	PT	AR	CPU ID
CATOJA	LNKEDT	11/17/1997	08:50:19	00:00:11	10	F4	002	444444
CATPRV	LNKEDT	11/18/1997	10:19:24	00:00:21	10	F7	002	333333
CATSDS	LNKEDT	11/18/1997	10:19:28	00:00:22	10	F8	002	333333
CICSEPIC	DTSANALS	11/18/1997	06:17:02	00:00:08	10	F2	002	222222
	IESWAITT	11/18/1997	06:17:10	00:00:14	10		000	
	DFHSIP	11/18/1997	06:17:25	02:30:32	17		000	
CICSEPIC	DTSANALS	11/18/1997	09:04:49	00:00:10	10	F2	000	222222
	IESWAITT	11/18/1997	09:05:00	00:00:04	10		000	
	DFHSIP	11/18/1997	09:05:04	00:06:37	17		000	
CICSEPIC	DTSANALS	11/18/1997	09:12:29	00:00:03	10	F2	000	222222
	IESWAITT	11/18/1997	09:12:33	00:00:01	10		000	
	DFHSIP	11/18/1997	09:12:34	00:10:39	17		000	
CICSEPIC	DTSANALS	11/18/1997	09:24:19	00:00:06	10	F2	000	222222
	IESWAITT	11/18/1997	09:24:26	00:00:02	10		000	
CICSEPI3	DTSANALS	11/17/1997	08:19:40	00:00:52	10	F2	000	444444
	IESWAITT	11/17/1997	08:20:33	00:00:53	10		000	
CICSICCF	DTSANALS	11/18/1997	06:07:49	00:00:03	10	F2	000	222222
	IESWAITT	11/18/1997	06:07:53	00:00:23	10		000	
	DFHSIP	11/18/1997	06:08:16	00:03:37	18		000	
CLEARRF	TSIDRFS	11/14/1997	10:47:16	00:00:01	10	F4	000	444444
PF3=END PF5=DETAIL PF7=BACKWARD PF8=FORWARD								

Fields

JOB NAME specifies the name of the job that initiated the open and close activity. Different jobs are separated from each other by a dotted line.

PHASE NAME is the name of the executed phase.

START DATE is the date the program started.

START TIME is the time the program started.

DURATION is the length of time the program ran.

CC is the VSE cancel code, if the program abended.

PT is the partition the job ran in.

AR is the number of audit records for the phase. There is one audit record for each open and one for each close processed by the phase.

CPU ID is the CPU the job ran in.

The Program Audit Panel

The Program Audit panel displays information about what datasets were processed by the program.

BIM-EPIC/VSE ONLINE JOB ACCOUNTING FACILITY									
FILEID	REC	CREATION	OPEN	EXPIRY	SERIAL	REEL	END	DTF	
PROCESSED	TYPE	TIME	CLOSE	CUU	DATE	ACCEPTED	TRACK	TRACK	TYPE

3F8.IJSYSLN	DO	10:19:29	C	400	11/18/1997	EPV400	030474	010000	32
3F8.IJSYSLN	DI	10:19:47	C	400	11/18/1997	EPV400	030474	010000	32

PF2=BACKWARD

PF4=END

Fields

FILEID PROCESSED is the name of the file processed. MT indicates that the phase was TSIDMNT and these records are DSN Catalog transactions.

REC TYPE is the record type of the file. Valid types are DO=disk output, DI=disk input, TO=tape output, TI=tape input, and MT>manual catalog maintenance transaction.

CREATION TIME is the time the dataset was created.

OPEN CLOSE indicates whether the file was opened or closed.

CUU is the device address accessed.

EXPIRY DATE is the projected expiration date of the dataset.

SERIAL ACCEPTED is the volume serial number if the **ACCEPT** command was issued.

REEL TRACK is the starting address of a disk dataset or the reel sequence number of a tape dataset.

END TRACK is the ending address of a disk dataset.

DTF TYPE is the DTF type of the dataset.

TSIDARP Report

Description

TSIDARP produces the Activity Log Report listing open and close activity by job name. Job data is listed in sections with information about the job and program information listed in the first line of the section. Dataset activity for that job is listed under the job name.

Fields

JOB NAME specifies the name of the job which initiated the open and close activity. Different jobs are separated from each other by a dotted line. PHASE NAME is the name of the program being reported.

Under each job and phase name, the Activity Log Report lists the datasets processed by each phase. It indicates what kind of dataset was processed, and whether the dataset was opened or closed.

PART is the partition the job ran in.

CPU TIME is the amount of CPU time the program used.

OVERHEAD is the program overhead, as defined by IBM.

GEN/REEL is the generation number of a disk version or the reel sequence of a tape version.

SERIAL NO is the serial number of the tape or disk pack accessed.

DTF TYPE is the dtf type of the dataset processed.

UNIT is the address of the drive used.

TOTAL I/O is the amount of I/O issued by the program to that device.

START/OPEN is the time (in HH.MM.SS format) the dataset was opened.

END/CLOSE is the time (in HH.MM.SS format) the dataset was closed.

USAGE MIN/SEC is the amount of time the dataset was processed.

D/S CREATION: JOB is the dataset's creation job.

D/S CREATION: DATE is the dataset's creation date.

DISK SPACE ALLOCAT: START is the beginning address of a disk extent.

DISK SPACE ALLOCAT: REQST is the amount of allocation requested for the extent.

DISK SPACE ALLOCAT: TRUNC is the extent size after truncation.

Activity Log Report Example

J O B A C C O U N T I N G S Y S T E M															
TSIDARP= 6.01.00															
C.P.U. NO. 111111	DATE 12/02/1997	A C T I V I T Y L O G													
JOB NAME	PHASE NAME	PART	C.P.U. TIME	OVER HEAD	GEN/ REEL	SERIAL NO.	DTF TYPE	UNIT	TOTAL I/O	START/ OPEN	END/ CLOSE	USAGE MIN/SEC	-D/S CREATION	DISK SPACE	ALLOCAT- REQST TRUNC
DUMMY	BG		00.1	00.1						09.30.15	09.30.17	.02	PROG TERMINATED	INTERR.HEX CODE 00	
BGINIT	IESWAIT	BG	00.3	00.2						09.29.40	09.30.10	.30			
VERIFYJ1	VERIFYJO	BG	00.7	00.5						09.56.19	09.56.39	.20			
	DISK OUTPUT		VERIFYJ.CONTROLLED.CAT												
			11			CKD001	20	160	91	09.56.31			VERIFYJ1 097336	10176	33 33
	CLOSED AS OUTPUT		VERIFYJ.CONTROLLED.CAT								09.56.35		VERIFYJ1 097336	10176	33 2
			11			CKD001	20	160							
VERIFYJ1	VERIFYJO	BG	00.7	00.5						09.57.47	09.58.05	.18			
	DISK OUTPUT		VERIFYJ.CONTROLLED.CAT												
			12			CKD001	20	160	91	09.57.58			VERIFYJ1 097336	10176	33 33
	CLOSED AS OUTPUT		VERIFYJ.CONTROLLED.CAT								09.58.02		VERIFYJ1 097336	10176	33 2
			12			CKD001	20	160							
VERIFYJ1	VERIFYJO	BG	00.7	00.5						10.01.52	10.02.16	.24			
VERIFYJ1	VERIFYJO	BG	00.7	00.5						10.03.40	10.04.05	.25			
VERIFYJ1	VERIFYJO	BG	00.6	00.5						10.09.02	10.09.19	.17			
	DISK OUTPUT		VERIFYJ.CONTROLLED.CAT												
			13			CKD001	20	160	70	10.08.57			VERIFYJ1 097336	10176	33 33
	CLOSED AS OUTPUT		VERIFYJ.CONTROLLED.CAT								10.09.01		VERIFYJ1 097336	10176	33 2
			13			CKD001	20	160							
	CLOSED AS INPUT		VERIFYJ.CONTROLLED.CAT								10.09.16		VERIFYJ1 097336	10176	33 2
			13			CKD001	20	160							
	DISK INPUT		VERIFYJ.CONTROLLED.CAT												
			13			CKD001	20	160	*	10.09.13			VERIFYJ1 097336	10176	33 2
VERIFYJ1	JCOPY	BG	00.7	00.3						10.08.53	10.09.02	.09			
VERIFYJ1	VERIFYJO	BG	00.6	00.5						10.10.17	10.10.31	.14			
	DISK OUTPUT		VERIFYJ.CONTROLLED.CAT												
			14			CKD001	20	160	70	10.10.13			VERIFYJ1 097336	10176	33 33
	CLOSED AS OUTPUT		VERIFYJ.CONTROLLED.CAT								10.10.15		VERIFYJ1 097336	10176	33 2
			14			CKD001	20	160							
	CLOSED AS INPUT		VERIFYJ.CONTROLLED.CAT								10.10.28		VERIFYJ1 097336	10176	33 2
			14			CKD001	20	160							
	DISK INPUT		VERIFYJ.CONTROLLED.CAT												
			14			CKD001	20	160	*	10.10.25			VERIFYJ1 097336	10176	33 2
VERIFYJ1	JCOPY	BG	00.7	00.3						10.10.10	10.10.16	.06			
VERIFYJ1	VERIFYJO	BG	00.5	00.5						10.16.11	10.16.25	.14			
VERIFYJ1	VERIFYJO	BG	00.5	00.4						10.14.33	10.14.55	.22			
VERIFYJ1	JCOPY	BG	00.6	00.3						10.15.59	10.16.10	.11			
* INDICATES THAT TOTAL I/O IS FOR MULTIPLE FILES IN SAME DEVICE															
J O B A C C O U N T I N G S Y S T E M															
TSIDARP= 6.01.00															
C.P.U. NO. 111111	DATE 12/02/1997	A C T I V I T Y L O G													
JOB NAME	PHASE NAME	PART	C.P.U. TIME	OVER HEAD	GEN/ REEL	SERIAL NO.	DTF TYPE	UNIT	TOTAL I/O	START/ OPEN	END/ CLOSE	USAGE MIN/SEC	-D/S CREATION	DISK SPACE	ALLOCAT- REQST TRUNC
VERIFYJ1	JCOPY	BG	00.6	00.3						10.14.20	10.14.33	.13			
TSIDRFS	TSIDRFS	F4	00.5	00.3						09.28.33	09.30.03	1.30			
JAY	JCOPY	F4	00.6	00.3						09.47.48	09.47.56	.08			
	DISK OUTPUT		VERIFYJ.CONTROLLED.CAT												
			9			CKD001	20	160	23	09.45.15			JAY	097336	10176 33 33
	DISK OUTPUT		VERIFYJ.CONTROLLED.CAT										JAY	097336	10176 33 33
			10			CKD001	20	160	*	09.46.44					
	DISK OUTPUT		VERIFYJ.IJSYSLS										JAY	097336	474 2000 2000
			18			EPV400	20	400	17	09.46.09					
	DISK OUTPUT		VERIFYJ.IJSYSLS										JAY	097336	474 2000 2000
			19			EPV400	20	400	*	09.47.29					
	CLOSED AS OUTPUT		VERIFYJ.CONTROLLED.CAT								09.45.20		JAY	097336	10176 33 2
			9			CKD001	20	160							
	CLOSED AS OUTPUT		VERIFYJ.CONTROLLED.CAT								09.46.51		JAY	097336	10176 33 2
			10			CKD001	20	160							
	CLOSED AS OUTPUT		VERIFYJ.IJSYSLS								09.46.13		JAY	097336	474 2000 416
			18			EPV400	20	400							
	CLOSED AS OUTPUT		VERIFYJ.IJSYSLS								09.47.32		JAY	097336	474 2000 416
			19			EPV400	20	400							
	CLOSED AS INPUT		VERIFYJ.CONTROLLED.CAT								09.45.36		JAY	097336	10176 33 2
			9			CKD001	20	160							
	CLOSED AS INPUT		VERIFYJ.CONTROLLED.CAT								09.47.05		JAY	097336	10176 33 2
			10			CKD001	20	160							
	CLOSED AS INPUT		VERIFYJ.IJSYSLS								09.46.28		JAY	097336	474 2000 416
			18			EPV400	20	400							

TSIDBRP Reports

Description	TSIDBRP creates the DSN/JOB Cross-Reference Report (a cross-reference to the Activity Log Report) listing open and close activity sequenced by dataset name. TSIDBRP also creates the Accepted Tapes Report which lists each time ACCEPT was used as a response to message EP008 or EP009.
Cross-Reference Fields	<p>Each dataset is reported with a separate line for each open and each close. The last column, which has no header, specifies whether the line refers to open or close activity and whether the file was used as input or output. "OPEN OUTPUT INPUT NEXT REEL EXTENT" indicates that an open was initiated for the next reel or extent in the sequence.</p> <p>FILE NAME is the name of the dataset opened or closed. JOB NAME is the name of the job which initiated the activity. PHASE NAME is the name of the program being reported. CPUID is the CPU the program ran in. RUN DATE is the date the job ran. TIME is the time the job started. PART is the partition the job ran in. SERIAL NO. is the volume serial number which was opened or closed. DEVICE: TYPE and NO. is the type of device used and the device address. REEL NO. is the reel or extent sequence number. START TRACK is the starting address of disk extents. NO. OF TRACKS is the number of tracks requested for the allocation. TRACKS TRUNCT. is the number of tracks after truncation.</p>

Cross-Reference Report Example

J O B A C C O U N T I N G S Y S T E M												TSIDBRP= 6.1					
DSN / JOB CROSS REFERENCE												RUN DATE 04/16/1998		TIME 13.16.35		PAGE 1	
FILE NAME	JOB NAME	PHASE NAME	CPUID	RUN DATE	TIME	PART	SERIAL NO.	-DEVICE- TYPE	REEL NO.	START TRACK	NO OF TRACKS	TRACKS TRUNCT.					
VERIFY6.FILE01	VERIFY6	JCOPY	G=0000002	222222 04/15/1998	06.19.25	BG	CKD001	DISK 160		9645	15	15	OPENED	OUTPUT			
			222222		06.19.31	BG	CKD001	DISK 160		9645	15	7	CLOSED	OUTPUT			
VERIFY6.FILE01	VERIFY6	JCOPY	G=0000003	222222 04/15/1998	06.19.35	BG	CKD001	DISK 160		9652	15	15	OPENED	OUTPUT			
			222222		06.19.44	BG	CKD001	DISK 160		9652	15	7	CLOSED	OUTPUT			
VERIFY6.FILE01	VERIFY6	JCOPY	G=0000004	222222 04/15/1998	06.19.47	BG	CKD001	DISK 160		9659	15	15	OPENED	OUTPUT			
			222222		06.19.54	BG	CKD001	DISK 160		9659	15	7	CLOSED	OUTPUT			
VERIFY6.FILE01	VERIFY6	JCOPY	G=0000005	222222 04/15/1998	06.20.00	BG	CKD001	DISK 160		9666	15	15	OPENED	OUTPUT			
			222222		06.20.06	BG	CKD001	DISK 160		9666	15	7	CLOSED	OUTPUT			
VERIFY6.FILE01	VERIFY6	JCOPY	G=0000006	222222 04/15/1998	06.20.10	BG	CKD001	DISK 160		9749	15	15	OPENED	OUTPUT			
			222222		06.20.18	BG	CKD001	DISK 160		9749	15	7	CLOSED	OUTPUT			
		TSIDOFI	222222		06.27.30	BG	CKD001	DISK 160		9749	15	7	OPENED	INPUT			
			222222		06.27.31	BG	CKD001	DISK 160		9749	15	7	CLOSED	INPUT			
VERIFY6.FILE01	VERIFY6	JCOPY	G=0000007	222222 04/15/1998	06.20.20	BG	CKD001	DISK 160		9756	15	15	OPENED	OUTPUT			
			222222		06.20.27	BG	CKD001	DISK 160		9756	15	7	CLOSED	OUTPUT			
		TSIDOFI	222222		06.27.32	BG	CKD001	DISK 160		9756	15	7	OPENED	INPUT			
			222222		06.27.33	BG	CKD001	DISK 160		9756	15	7	CLOSED	INPUT			
VERIFY6.FILE01	VERIFY6	JCOPY	G=0000008	222222 04/15/1998	06.20.29	BG	CKD001	DISK 160		9763	15	15	OPENED	OUTPUT			
			222222		06.20.35	BG	CKD001	DISK 160		9763	15	7	CLOSED	OUTPUT			
		TSIDOFI	222222		06.27.33	BG	CKD001	DISK 160		9763	15	7	OPENED	INPUT			
			222222		06.27.34	BG	CKD001	DISK 160		9763	15	7	CLOSED	INPUT			
VERIFY6.FILE01	VERIFY6	JCOPY	G=0000009	222222 04/15/1998	06.20.37	BG	CKD001	DISK 160		9770	15	15	OPENED	OUTPUT			
			222222		06.20.44	BG	CKD001	DISK 160		9770	15	7	CLOSED	OUTPUT			
VERIFY6.FILE01	VERIFY6	TSIDONL	G=0000010	222222 04/15/1998	06.28.11	BG	CKD001	DISK 160		9787	15	15	OPENED	OUTPUT			
			222222		06.28.14	BG	CKD001	DISK 160		9787	15	7	CLOSED	OUTPUT			
VERIFY6.FILE02	VERIFY6	JCOPY	G=0000002	222222 04/15/1998	06.20.48	BG	CKD001	DISK 160		9673	5	5	OPENED	OUTPUT			
			222222		06.20.55	BG	CKD001	DISK 160		9678	2	2	OUTPUT	NEXT EXTENT			
			222222		06.20.56	BG	CKD001	DISK 160		9678	2	2	CLOSED	OUTPUT			
		TSIDOFI	222222		06.27.16	BG	CKD001	DISK 160		9673	5	5	OPENED	INPUT			
			222222		06.27.16	BG	CKD001	DISK 160		9678	2	2	OPENED	INPUT			
			222222		06.27.17	BG	CKD001	DISK 160		9673	5	5	CLOSED	INPUT			
VERIFY6.FILE02	VERIFY6	JCOPY	G=0000003	222222 04/15/1998	06.20.59	BG	CKD001	DISK 160		9777	5	5	OPENED	OUTPUT			
			222222		06.20.59	BG	CKD001	DISK 160		9777	5	5	OPENED	OUTPUT			
		TSIDOFI	222222		06.21.04	BG	CKD001	DISK 160		9680	2	2	OUTPUT	NEXT EXTENT			
			222222		06.21.06	BG	CKD001	DISK 160		9680	2	2	CLOSED	OUTPUT			
			222222		06.27.18	BG	CKD001	DISK 160		9777	5	5	OPENED	INPUT			
			222222		06.27.19	BG	CKD001	DISK 160		9777	5	5	CLOSED	INPUT			
			222222		06.27.19	BG	CKD001	DISK 160		9680	2	2	OPENED	INPUT			
VERIFY6.FILE02	VERIFY6	JCOPY	G=0000004	222222 04/15/1998	06.21.08	BG	CKD001	DISK 160		9782	5	5	OPENED	OUTPUT			
			222222		06.21.14	BG	CKD001	DISK 160		9682	2	2	OUTPUT	NEXT EXTENT			
			222222		06.21.16	BG	CKD001	DISK 160		9682	2	2	CLOSED	OUTPUT			
		TSIDOFI	222222		06.27.20	BG	CKD001	DISK 160		9782	5	5	OPENED	INPUT			
			222222		06.27.21	BG	CKD001	DISK 160		9682	2	2	OPENED	INPUT			
			222222		06.27.22	BG	CKD001	DISK 160		9782	5	5	CLOSED	INPUT			
VERIFY6.FILE02	VERIFY6	JCOPY	G=0000005	222222 04/15/1998	06.21.18	BG	CKD001	DISK 160		9787	5	5	OPENED	OUTPUT			
			222222		06.21.24	BG	CKD001	DISK 160		9792	2	2	OUTPUT	NEXT EXTENT			
			222222		06.21.25	BG	CKD001	DISK 160		9792	2	2	CLOSED	OUTPUT			
		TSIDOFI	222222		06.27.23	BG	CKD001	DISK 160		9787	5	5	OPENED	INPUT			
			222222		06.27.24	BG	CKD001	DISK 160		9787	5	5	CLOSED	INPUT			
			222222		06.27.24	BG	CKD001	DISK 160		9792	2	2	OPENED	INPUT			

Accepted Tapes Report Fields

REQUESTED FILE NAME is the name of the dataset requested by the job.

REQUESTED SERIAL NO. is the volume serial number requested by the job. A serial number of '000000' indicates that no real volume number is known.

ACCEPTED FILE NAME is the name of the dataset accepted.

ACCEPTED SERIAL NO. is the volume serial number accepted. A serial number of '000000' indicates that no real volume number is known.

JOB NAME is the job that was running when the operator accepted the tape.

PHASE NAME is the phase that was running when the operator accepted the tape.

DATE is the date the tape was accepted.

TIME is the time the tape was accepted.

PART is the partition the job was running in.

Accepted Tapes Report Example

J O B A C C O U N T I N G S Y S T E M				TSIDBRP= 6.1					
				ACCEPTED TAPES REPORT		RUN DATE 12/30/1997		TIME 07.28.58	PAGE 1
-----REQUESTED-----		-----ACCEPTED-----							
FILE NAME	SERIAL NO.	FILE NAME	SERIAL NO.	JOB NAME	PHASE NAME	DATE	TIME	PART	
VERIFY4.EMPTY	000000	*** UNLABELED ***	000000	VERIFY41	JCOPY	12/30/1997	06.16.42	U1	
VERIFY4.EMPTY	000000	VERIFY4.UNCONTROL	VRFY41	VERIFY41	JCOPY	12/30/1997	06.16.59	U1	
VERIFY4.TAPE	VRFY42	VERIFY4.UNCONTROL	VRFY41	VERIFY41	JCOPY	12/30/1997	06.17.33	U1	
VERIFY4.EMPTY	000000	VERIFY4.TAPE1	VRFY41	VERIFY41	JCOPY	12/30/1997	06.17.56	U1	
VERIFY4.TAPE2	VRFY42	VERIFY4.TAPE1	VRFY41	VERIFY41	JCOPY	12/30/1997	06.18.05	U1	
VERIFY4.TAPE2	VRFY42	VERIFY4.TAPE1	VRFY41	VERIFY41	JCOPY	12/30/1997	06.18.18	U1	
VERIFY4.SEVENTEEN	VRFY41	VERIFY4.MORE.THAN	VRFY42	VERIFY41	JCOPY	12/30/1997	06.18.47	U1	
VERIFY4.MORE.THAN	VRFY42	VERIFY4.SEVENTEEN	VRFY41	VERIFY41	JCOPY	12/30/1997	06.18.59	U1	
VERIFY4.SEVENTEEN	VRFY41	VERIFY4.MORE.THAN	VRFY42	VERIFY41	JCOPY	12/30/1997	06.19.13	U1	
VERIFY4.MORE.THAN	VRFY42	VERIFY4.SEVENTEEN	VRFY41	VERIFY41	JCOPY	12/30/1997	06.19.25	U1	

TSIDDIS Report

Description

TSIDDIS produces a summary report correlating dataset versions and tape volume serial numbers. It is listed in dataset name order. There is one entry for each dataset version. If a version spans more than one volume, there is an entry for each volume.

For information on creating TSIDDIS reports, see 12-24 in this manual.

Fields

DATASET NAME indicates the dataset name. Versions and reels listed between this name and the next belong to this dataset name.

VER is the relative version number of the version.

REEL is the reel sequence number associated with the tape (important for multi-volume versions).

SER NO is the volume serial number of the tape associated with the version.

TSIDDIS Report Example

TSIDDIS= 6.01.00				DATA SET MANAGEMENT SYSTEM				PAGE 1			
TIME 14 06 30								DATE 12/04/1997			
DATA SET NAME	VER	REEL	SER NO	DATA SET NAME	VER	REEL	SER NO				
\$\$\$VERIFY8.JUAN	***	EMPTY	***	CI.PROD.NCL.EL511WA.MTH511.ENTRY	1	1	012086				
A-R TAPE	1	1	001249	CI.PROD.NCL.EL511WA.VNB511.ENTRY	1	1	011950				
	2	1	000114		2	1	012093				
ALERT.AUDIT.HIST	1	1	000092	CI.PROD.NCL.EL513WA.EL513.ENTRY	1	1	011635				
	2	1	000057		2	1	011989				
	3	1	000455		3	1	011966				
AR.PROD.LGX.AR192BKP.GVEXPORT.AR192RST	1	1	011609		4	1	011954				
	2	1	011918		5	1	011859				
	3	1	012056		6	1	012130				
AR.PROD.LGX.EL588BKP.GVEXPORT.EL588RST	1	1	011615		7	1	012122				
	2	1	011932		8	1	012105				
	3	1	012087		9	1	012085				
AR.PROD.NCL.CASHRCT.CASHRBKP.DAILY	1	1	011610		10	1	012099				
	2	1	011904	CI.PROD.NCL.E001WA.P4107WRK.JE004RPT	1	1	010947				
	3	1	012057		2	1	012111				
AR.PROD.NCL.ELME3BKP.GVEXPORT.ELME3RST	1	1	010523		3	1	010108				
	2	1	011185		4	1	011489				
	3	1	011495		5	1	010189				
AR.PROD.NCL.ELME3BKP.GVEXPORT.OFFSITE	1	1	090783		6	1	011560				
	2	1	090651		7	1	011559				
	3	1	090529	CI.TEST.LGX.EL512BKP.GVEXPORT.EL512RST	1	1	011498				
	4	1	090424		2	1	012223				
	5	1	090217	CI.TEST.NCL.EL511WA.CI003CNV.ENTRY	1	1	010464				
	6	1	090218	CI.TEST.NCL.EL511WA.CI005CNV.ENTRY	1	1	011505				
	7	1	090123		2	1	011496				
AR.PROD.NCL.ERBFWD.EL863.MONTHEND	1	1	090122		3	1	012226				
AR.PROD.NCL.EREXTR.EL860.MONTHEND	1	1	010405		4	1	012236				
	1	1	012145	CI.TEST.NCL.EL511WA.VNB511.ENTRY	1	1	011521				
	2	1	012062	CI.TEST.NCL.EL513WA.EL513.ENTRY	1	1	011494				
	3	1	010340		2	1	012228				
AR.PROD.NCL.EREXTR.EL860.OFFSITE	1	1	090785	CL.PROD.LGX.CLDLYBKP.GVEXPORT.CLDLYRST	1	1	011621				
	2	1	090653		2	1	011629				
	3	1	090531		2	1	011961				
	4	1	090425		2	1	011947				
	5	1	090209		3	1	012135				
	6	1	090149		2	1	012089				
AR.PROD.NCL.ERPND.EL860.MONTHEND	7	1	090126	CL.PROD.LGX.CLEOMBKP.GVEXPORT.CLEOMRST	1	1	011530				
	1	1	012149		2	1	011516				
	2	1	012067	CL.PROD.LGX.CL311BKP.GVEXPORT.CL311RST	1	1	011623				
	3	1	010351		2	1	011644				
AR.PROD.NCL.ERPND.EL860.OFFSITE	1	1	090786		2	1	011968				
	2	1	090654		2	1	011969				
	3	1	090532		3	1	012097				
	4	1	090426		2	1	012139				
	5	1	090208	CL.PROD.LGX.ELEXTR.CLEXTBKP.CLDLY	1	1	011637				
	6	1	090128		2	1	011974				
	7	1	090127		3	1	012136				
AR.PROD.NCL.ERPYAJ.EL860.MONTHEND	1	1	012147		4	1	010787				
	2	1	012038		5	1	010767				
	3	1	010355		6	1	011322				
AR.PROD.NCL.ERPYAJ.EL860.OFFSITE	1	1	090880		7	1	010220				
	2	1	090784	CL.PROD.LGX.ELEXTR.CLEXTBKP.OFFSITE	1	1	090765				
	3	1	090652		2	1	090635				
	4	1	090530		3	1	090491				
	5	1	090423		4	1	090340				
	6	1	090210		5	1	090156				
	7	1	090125		6	1	090102				
	8	1	090124		7	1	090101				
BILRECS BKUP TAPE	1	1	000050	CL.PROD.LGX.ELHAF.CLHAFPCPY.MONTHEND	1	1	011393				
	2	1	000024		2	1	011395				
	3	1	000061		3	1	011397				
	4	1	000019		2	1	011865				
	5	1	000143		2	1	011855				
	6	1	000305		3	1	011867				
	7	1	000035	CL.PROD.LGX.ELHAF.EL309.MONTHEND	1	1	010715				
BUD.ACASH.BKUP	1	1	000182		2	1	010716				
BUDGET.SQLBKUP	1	1	001056		3	1	010718				
	2	1	000252		2	1	011861				
	3	1	000432		2	1	011854				
BUDGET.SQLBKUP1	1	1	001056		3	1	011864				
	2	1	000252		3	1	010798				
	3	1	000432		2	1	010804				
CASH.BUD1.BKUP	1	1	000149		3	1	010839				
CASH.BUD2.BKUP	1	1	000149	CL.PROD.LGX.EL000BKP.CLEOMREN.OFFSITE	1	1	090769				
CI.PROD.LGX.EL512BKP.GVEXPORT.EL512RST	1	1	011625		2	1	090770				
	2	1	011967		3	1	090771				
	3	1	011953		4	1	090772				
	4	1	011938		5	1	090773				
	5	1	011850		6	1	090774				
	6	1	011851		7	1	090775				

TSIDJAC Report

Description

TSIDJAC creates the Job Time Activity Report, which lists job accounting information by CPU and by job.

Fields

CPU NO. is the CPU being reported.
JOB NAME is the name of the job which initiated the activity.
RUN DATE is the date the job ran.
PHASE NAME is the name of the program being reported.
PART ID is the partition the job ran in.
START TIME is the time (in hh.mm.ss format) the phase started.
END TIME is the time the phase completed.
DURATION TIME is the length of time the phase ran.
CPU TIME is the amount of CPU time the program used.
OVERHEAD is the program overhead, as defined by IBM.
TOTAL I/O is the amount of I/O issued by the program to that device.

In addition, grand totals for DURATION TIME, CPU TIME, OVERHEAD, and TOTAL I/O are listed at the end of the report.

Job Time Activity Report Example

C.P.U. NO. 111111				* J O B A C C O U N T I N G S Y S T E M			TIME 13.20.07		TSIDJAC= 6.01.00	
JOB NAME	RUN DATE	PHASE NAME	PART ID	J O B T I M E S T A R T	J O B T I M E E N D	J O B T I M E D U R A T I O N	CPU TIME	OVER HEAD	PAGE TOTAL I/O	1
BGINIT	12/02/1997	IESWAIT	BG	09.29.40	09.30.10	00.00.30 00.00.30* 00.00.30**	00.00.3 00.00.3* 00.00.3**	00.00.2 00.00.2* 00.00.2**	238 238* 238**	
DUMMY	12/02/1997		BG	09.30.15	09.30.17	00.00.02 00.00.02* 00.00.02**	00.00.1 00.00.1* 00.00.1**	00.00.1 00.00.1* 00.00.1**	44 44* 44**	
DYNBGN1	12/02/1997		Z1	09.28.33	09.30.29	00.01.56 00.01.56* 00.01.56**	00.01.0 00.01.0* 00.01.0**	00.00.4 00.00.4* 00.00.4**	191 191* 191**	
F6BGN	12/02/1997		F6	09.27.56	09.30.07	00.02.11 00.02.11* 00.02.11**	00.01.1 00.01.1* 00.01.1**	00.00.5 00.00.5* 00.00.5**	189 189* 189**	
F7BGN	12/02/1997		F7	09.27.58	09.30.14	00.02.16 00.02.16* 00.02.16**	00.01.1 00.01.1* 00.01.1**	00.00.5 00.00.5* 00.00.5**	189 189* 189**	
F8BGN	12/02/1997		F8	09.27.59	09.30.13	00.02.14 00.02.14* 00.02.14**	00.01.0 00.01.0* 00.01.0**	00.00.5 00.00.5* 00.00.5**	189 189* 189**	
F9BGN	12/02/1997		F9	09.28.01	09.30.16	00.02.15 00.02.15* 00.02.15**	00.01.1 00.01.1* 00.01.1**	00.00.5 00.00.5* 00.00.5**	189 189* 189**	
JAY	12/02/1997	JCOPY	F4	09.44.59	09.45.22	00.00.23	00.01.1	00.00.7	705	
	12/02/1997	VERIFYJI	F4	09.45.22	09.45.39	00.00.17	00.00.5	00.00.5	524	
	12/02/1997	JCOPY	F4	09.45.39	09.45.45	00.00.06	00.00.8	00.00.3	336	
	12/02/1997	TSIDLVT	F4	09.45.46	09.46.15	00.00.29	00.01.7	00.00.7	696	
	12/02/1997	JCOPY	F4	09.46.15	09.46.31	00.00.16	00.01.1	00.00.4	1448	
	12/02/1997	JCOPY	F4	09.46.32	09.46.38	00.00.06	00.00.6	00.00.3	245	
	12/02/1997	JCOPY	F4	09.46.38	09.46.53	00.00.15	00.00.6	00.00.4	357	
	12/02/1997	VERIFYJI	F4	09.46.54	09.47.07	00.00.13	00.00.5	00.00.5	521	
	12/02/1997	JCOPY	F4	09.47.07	09.47.14	00.00.07	00.00.7	00.00.3	336	
	12/02/1997	TSIDLVT	F4	09.47.14	09.47.33	00.00.19	00.01.7	00.00.7	684	
	12/02/1997	JCOPY	F4	09.47.33	09.47.47	00.00.14	00.01.1	00.00.4	1457	
	12/02/1997	JCOPY	F4	09.47.48	09.47.56	00.00.08	00.00.6	00.00.3	400	
	12/02/1997	JCOPY	F4	09.50.11	09.50.31	00.00.20	00.01.0	00.00.6	635	
	12/02/1997	VERIFYJI	F4	09.50.32	09.50.44	00.00.12	00.00.5	00.00.5	485	
	12/02/1997	TSIDLVT	F4	09.50.45	09.51.02	00.00.17	00.01.7	00.00.7	684	
	12/02/1997	JCOPY	F4	09.51.03	09.51.07	00.00.04	00.01.0	00.00.4	1446	
	12/02/1997	JCOPY	F4	09.51.08	09.51.11	00.00.03	00.00.5	00.00.3	248	
	12/02/1997	JCOPY	F4	09.51.11	09.51.17	00.00.06	00.00.6	00.00.3	284	
	12/02/1997	VERIFYJI	F4	09.51.17	09.51.31	00.00.14	00.00.5	00.00.5	487	
	12/02/1997	TSIDLVT	F4	09.51.32	09.51.54	00.00.22	00.01.7	00.00.7	695	
	12/02/1997	JCOPY	F4	09.51.54	09.52.06	00.00.12	00.01.1	00.00.4	1446	
	12/02/1997	JCOPY	F4	09.52.07	09.52.15	00.00.08	00.00.6	00.00.3	404	
	12/02/1997	JCOPY	F4	09.56.03	09.56.46	00.00.43	00.00.8	00.00.5	595	
	12/02/1997	JCOPY	F4	09.56.46	09.56.52	00.00.06	00.00.7	00.00.2	336	
	12/02/1997	TSIDLVT	F4	09.56.53	09.57.16	00.00.23	00.01.7	00.00.7	684	
	12/02/1997	JCOPY	F4	09.57.16	09.57.35	00.00.19	00.01.1	00.00.4	1457	
	12/02/1997	JCOPY	F4	09.57.35	09.57.43	00.00.08	00.00.6	00.00.3	244	
	12/02/1997	JCOPY	F4	09.57.44	09.58.10	00.00.26	00.00.4	00.00.2	251	
	12/02/1997	JCOPY	F4	09.58.10	09.58.15	00.00.05	00.00.7	00.00.2	336	
	12/02/1997	TSIDLVT	F4	09.58.16	09.58.34	00.00.18	00.01.7	00.00.8	686	
	12/02/1997	JCOPY	F4	09.58.35	09.58.48	00.00.13	00.01.1	00.00.4	1446	
	12/02/1997	JCOPY	F4	09.58.48	09.58.55	00.00.07	00.00.6	00.00.3	389	
	12/02/1997	JCOPY	F4	10.01.37	10.02.23	00.00.46	00.00.9	00.00.5	613	
	12/02/1997	TSIDLVT	F4	10.02.24	10.02.58	00.00.34	00.01.7	00.00.7	684	
	12/02/1997	JCOPY	F4	10.02.58	10.03.23	00.00.25	00.01.1	00.00.4	1457	
	12/02/1997	JCOPY	F4	10.03.24	10.03.34	00.00.10	00.00.5	00.00.3	242	
	12/02/1997	JCOPY	F4	10.03.34	10.04.11	00.00.37	00.00.4	00.00.2	269	
	12/02/1997	TSIDLVT	F4	10.04.11	10.04.29	00.00.18	00.01.7	00.00.7	684	
	12/02/1997	JCOPY	F4	10.04.30	10.04.42	00.00.12	00.01.1	00.00.4	1455	
	12/02/1997	JCOPY	F4	10.04.42	10.04.50	00.00.08	00.00.6	00.00.3	389	
	12/02/1997	JCOPY	F4	10.08.41	10.09.22	00.00.41	00.00.9	00.00.5	600	
	12/02/1997	JCOPY	F4	10.09.23	10.09.27	00.00.04	00.00.7	00.00.2	336	
	12/02/1997	TSIDLVT	F4	10.09.28	10.09.45	00.00.17	00.01.7	00.00.7	684	
	12/02/1997	JCOPY	F4	10.09.45	10.09.57	00.00.12	00.01.1	00.00.4	1446	
	12/02/1997	JCOPY	F4	10.09.58	10.10.06	00.00.08	00.00.5	00.00.3	244	
	12/02/1997	JCOPY	F4	10.10.06	10.10.35	00.00.29	00.00.4	00.00.2	256	
	12/02/1997	JCOPY	F4	10.10.35	10.10.37	00.00.02	00.00.7	00.00.2	336	
	12/02/1997	TSIDLVT	F4	10.10.38	10.10.48	00.00.10	00.01.7	00.00.7	693	
	12/02/1997	JCOPY	F4	10.10.48	10.10.56	00.00.08	00.01.1	00.00.4	1446	
	12/02/1997	JCOPY	F4	10.10.56	10.11.09	00.00.13	00.00.6	00.00.3	398	
	12/02/1997	JCOPY	F4	10.14.05	10.15.01	00.00.56	00.00.9	00.00.5	619	

TSIDLVT Report

Description

TSIDLVT lists VTOC data in batch. The report format varies depending on the options you choose, and the following report only a sample of the reports you can create.

For information on producing TSIDLVT reports, see 13-8 in this manual.

Fields

DISPLAY OF identifies the content of the report, either SER=volser or the name of the pool.

FILE NAME is the name of the dataset residing in the specified pools of the VTOC.

SERIAL is the volume serial number the extent resides on.

EXT SEQ is the extent sequence number.

EXTENT START is the starting extent address.

EXTENT END is the ending extent address.

EXTENT NUMBER specifies the number of extents occupied by the dataset.

CREATE DATE is the date the dataset was created.

EXPIRE DATE is the expiration date written on the VTOC.

DISK TYPE is the type of disk device.

FILE TYPE is the dataset's file type.

EXT If printed in the far right column, the extent has expired.

TSIDLVT Report Example

REPORT FOR SERIAL NUMBER=CKD001												
OPTIONS IN EFFECT - PRINT IN EXT SEQUENCE												
LIST EXPIRED EXTENTS												
IGNORE POOL LIMITS												
LIST FREE SPACE												
LIST IN RELATIVE TRACK/BLOCK FORMAT												
USE CONVENTIONAL DATE FORMAT MM/DD/YYYY												
LIST UNEXPIRED EXTENTS												

DATE 12/11/1997	DISPLAY OF	E P I C	V T O C	R E P O R T				VOLSER CKD001	TSIDLVT 6.1			
TIME 14 30 16	SER=CKD001	S E R I A L	I N	E X T E N T	S E Q U E N C E		CUU 160		PAGE			
	FILE NAME	SERIAL	EXT SEQ	START	EXTENT END	NUMBER	CREATE DATE	EXPIRE DATE	DISK TYPE	FILE TYPE		
VSE.POWER.QUEUE.CKD.FILE		CKD001	0	1	4	4	09/18/1997	12/31/1999	3390	DA		
***** FREE SPACE *****		CKD001		5	14	10			3390			
BIMEDIT.CKD.BIFLIB		CKD001	0	15	674	660	06/24/1997	12/31/1999	3390	DA		
Z9999994.VSAMDSPC.TAE74BEB.T03FDC09		CKD001	0	675	689	15	04/03/1997	12/32/1999	3390	VSAM		
Z9999992.VSAMDSPC.TAE74BEB.T5C0E97E		CKD001	0	690	6689	6000	04/03/1997	12/32/1999	3390	VSAM		
VSE.POWER.DATA.CKD.FILE		CKD001	0	6690	7139	450	09/18/1997	12/31/1999	3390	DA		
VSE.BIM.COLUMBUS.LIB6		CKD001	0	7140	8264	1125	07/11/1997	12/31/2099	3390	SD		
VSE.BIM.COLUMBUS.MON.PDS		CKD001	0	8265	8534	270	07/10/1997	12/31/2099	3390	SD		
VSE.BIM.COLUMBUS.VIO.PDS		CKD001	0	8535	9134	600	07/10/1997	12/31/2099	3390	SD		
VSE.BIM.COLUMBUS.EVT.PDS		CKD001	0	9135	9314	180	07/11/1997	12/31/2099	3390	SD		
VSE.BIM.COLUMBUS.LOG.PDS		CKD001	0	9315	9374	60	07/10/1997	12/31/2099	3390	SD		
VSE.BIM.COLUMBUS.CPR.PDS		CKD001	0	9375	9464	90	07/10/1997	12/31/2099	3390	SD		
VSE.BIM.COLUMBUS.ARC.PDS		CKD001	0	9465	9509	45	07/10/1997	12/31/2099	3390	SD		
EAK294.RECORDER.DATA		CKD001	1	9510	9521	12	12/04/1997	12/11/1997	3390	SD	EXP	
***** FREE SPACE *****		CKD001		9522	9549	28			3390			
I17007.RECORDER.FILE.BACKUP		CKD001	1	9550	9558	9	08/06/1997	12/31/2099	3390	SD		
I17007.CATALOG.FASTBU		CKD001	1	9559	9709	151	08/06/1997	12/31/2099	3390	SD		
F43.TSIDUTL.WORK.FILE		CKD001	1	9710	9711	2	12/04/1997	12/11/1997	3390	SD	EXP	
***** FREE SPACE *****		CKD001		9712	10173	462			3390			
HEWTEST1.Y2000.Y2000	G=0000002	CKD001	1	10174	10175	2	11/14/1997	12/31/2099	3390	SD		
***** FREE SPACE *****		CKD001		10176	10214	39			3390			
VSE.BIM.COLUMBUS.LIBRARY		CKD001	0	10215	12314	2100	09/10/1997	12/31/1999	3390	SD		
***** FREE SPACE *****		CKD001		12315	13058	744			3390			
HEW.RECORDER.FILE.BU.Y2000		CKD001	1	13059	13060	2	11/14/1997	12/14/1997	3390	SD		
***** FREE SPACE *****		CKD001		13061	13107	47			3390			
HEW.RECORDER.FILE.BU.Y2000.BIG		CKD001	1	13108	13119	12	11/18/1997	12/18/1997	3390	SD		
***** FREE SPACE *****		CKD001		13120	13278	159			3390			
JUANTEST	G=0000011	CKD001	1	13279	13287	9	11/25/1997	12/31/2099	3390	SD		
JUANTEST	G=0000012	CKD001	1	13288	13296	9	11/25/1997	12/31/2099	3390	SD		
JUANTEST	G=0000013	CKD001	1	13297	13306	10	11/25/1997	12/31/2099	3390	SD		
JUANTEST	G=0000014	CKD001	1	13307	13315	9	11/25/1997	12/31/2099	3390	SD		
JUANTEST	G=0000015	CKD001	1	13316	13324	9	11/25/1997	12/31/2099	3390	SD		
JUANTEST	G=0000016	CKD001	1	13325	13333	9	11/25/1997	12/31/2099	3390	SD		
JUANTEST	G=0000017	CKD001	1	13334	13342	9	11/25/1997	12/31/2099	3390	SD		
JUANTEST	G=0000018	CKD001	1	13343	13351	9	11/25/1997	12/31/2099	3390	SD		
JUANTEST	G=0000019	CKD001	1	13352	13360	9	11/25/1997	12/31/2099	3390	SD		
***** FREE SPACE *****		CKD001		13361	16678	3318			3390			
CKD001.QDAS.REL4.SHARE.FILE		CKD001	0	16679	16679	1	11/12/1997	12/31/2099	3390	SD		
VTOC		CKD001	0	16680	16694	15			3390	UN		

** VOLUME STATISTICS **												

	TOTAL SPACE THIS VOLUME	-			16695							
	TOTAL USED SPACE THIS VOLUME	-			11874							
	TOTAL FREE SPACE THIS VOLUME	-			4821							

TSIDMNT Reports

Description

TSIDMNT can produce a summary report or a detail report. To receive a summary report, request a display of all datasets or a group of datasets. To receive a detail report, request a specific dataset name or a specific volume serial number.

For information on creating TSIDMNT reports, see Chapter 3.

Summary Report

DATASET NAME is the name of the dataset.

SERIAL is the volume serial number(s) associated with the version.

VN is the version number.

RL is the reel or extent sequence number.

ST is the version's status. Valid status codes are:

- (blank)=active version
- C=conditional catalog status
- L=active version in a locked status
- M=MCAT version
- O=open status
- S=scratch status.

BLK is the block size of the version.

REC is the record size of the version.

TYPE TAPE and DISK indicate the storage medium of the version. EMPTY means there are no versions of this dataset.

Detail Report

The first three lines for each dataset report the dataset's defaults. Lower lines present the version level data for the dataset.

BIM-EPIC Dataset Definition (EDD)

The first field at the top left corner is the dataset name.

RET is the number of days each version is to be retained.

CYC is the number of versions (cycles) to be maintained.

VLT is the number of the vault method assigned to the dataset.

WORK identifies a work dataset.

password is the dataset's Unlock/Release password is displayed at the far right.

REBLK is the default block size.

POOL The disk pool name.

EXT=n is the first value specified is the size of the primary extent. The second value is the size of all secondary extents.

SEC is the maximum number of secondary extents for a version.

ALR is the number of records for allocation by logical records.

SID is the system ID.

UID is the user ID.

(UN)LBL indicates whether this is a standard label tape dataset.

MOD is the default tape density.

SYSnnn is the SYS number assigned to the dataset.
REC is the actual record size of the current version
BLK is the actual block size of the current version
REV indicates whether this is a revolving tape dataset.
COM is the user comment.
TPA= is the output default for TPA devices (YES|NO)
TPL is the tape pool ID.

Version Information

SER-NO is the tape or disk volume serial number associated with this version.
VN is the version number.
RL is the reel or extent sequence number.
CREDT is the creation date in Julian format.
EXPDT is the projected expiration date.
CREAT/JB is the name of the creating job.
CUU is the drive address the version was created on.
ST is the version's status. The codes are:

- C=conditional catalog status
- L=active version in a locked status
- M=MCAT version
- O=open status
- S=scratch status

LAST/JOB is the last job that accessed this version.
LASTDT is the last date this version was accessed.
USE is the number of times the tape was opened since the last clean date.
ERR is the number of read/write errors encountered since the last clean date.
CLN-DAT is the last clean date of this tape.
VLT is the current vault location of this tape.
SL is the current slot location of this tape.

TSIDOFD Report

Description

TSIDOFD produces a report listing the datasets to be offloaded or backed up. To produce this report, see 13-17 in this manual.

Fields

DATASET NAME is the name of the dataset to offload or backup.
 VER is the version number of the dataset to offload or backup.
 ALL VER indicates all versions were selected.
 UID indicates whether the dataset was selected by user ID.
 SID indicates whether the dataset was selected by system ID.
 VOLUME indicates whether the dataset was selected by volume.
 POOL ID indicates whether the dataset was selected by pool name.
 S/DLA indicates whether the dataset was selected by SLA or DLA.
 NEW indicates that only new datasets were selected.

Dataset Selection Report Example

TSIDOFD= 6.1		D A T A S E T S E L E C T I O N R E P O R T								PAGE 1	
TIME: 15.32.06		64,466=SELECTED ENTRY TABLE SIZE USED FOR THIS PROCESSING								DATE 12/11/1997	
INCLUDE PARAMETERS		DATA SET NAME		/ VER		ALLVER UID SID		VOLUME		POOLID S/DLA MISC	
OMIT PARAMETERS		DATA SET NAME		/ VER		ALLVER UID SID		VOLUME		POOLID	
						0001					
						VU					
THE FOLLOWING DATA SETS HAVE BEEN SELECTED											
DATA SET NAME				VER		DATA SET NAME				VER	
ABSENTEE.TEMP				1		ACCOUNTS.PAYABLE.FILE				1	
ACCTS.PAYABLE.TWO				1		AR CONDENSE FILE				1	
BATCH CONVERT OUT				1		BILL BACKUP				1	
BILL.REPRINT.FILE				1		BILLING.PAN.ORDER				1	
BLS.REC				1		COLL.TAPE				1	
COLLECT JOURNAL				1		CONST.FILE.OUT				1	
CONSTR.JOURNAL.FILE				1		DAILY.ARFILE.BKUP				1	
DAILY.BDTRNMT.BKUP				1		DAILY.BD2CES.BKUP				1	
DAILY.BILLING				1		DAILY.OVERPAYMENTS				1	
DB.MONTHLY.TRANS2				1		DBLABOR.DISTRIBUTION				1	
DCDCASTO.SORT				1		DCDMAINT.INPUT				1	
DCDRATXT.OUTPUT				1		DEFER.BILL.BKUP				1	
DFR.BILLNG.BKP.TAPE				1		DISTRIBUTION.FILE				1	
DITTO.DISK				1		DR-CR.OUT				1	
ELMET.INVENTORY.FILE				1		EPIC.TSIDOFL.INPUT				1	
EREP.HISTORY				1		EREP.HISTORY.OUT				1	
EXPANDED.NOTICE2				1		FA.CICS170.MSGUSR				1	
FINAL ENTREX				1		FINLODE.BKUP				1	
FINLODE.562				1		FSILODE.BKUP				1	
FSOHOST.BKUP				1		FUELLOD.FILE				1	
F8.CICS170.MSGUSR				1		GROUP1.DISKOC				1	
GROUP1.DISKOR				1		G7.BILLS				1	
HEWTEST1.Y2000.Y2000				1		HP TRAN HIST				1	
HPL.WORK.FILE				1		IND.AUDIT.FILE				1	
INVENTORY.CARD.FILE				1		JOB.ACCT.WORK				1	
JOB.HIST.ACCUM				1		JOB.HIST.DAILY				1	
JUANTEST				1		LTD.CONSTR.MASTER				1	
M & S.FILE				1		MAINT.REPORTT				1	
MD.DUMMY.FILE.G30026				1		MD.DUMMY.FILE.G30027				1	
MD.EXTRACT.CHECK.FILE				1		MD.GL.RECON.WRAP.FILE				1	
MD.HRM.ADJUST.FILE				1		MD.HRM.DISTR.INTRF				1	
MD.HRM.PAY070.PUNCH1				1		MD.HRM.PAY070.PUNCH10				1	
MD.HRM.PAY070.PUNCH2				1		MD.HRM.PAY070.PUNCH3				1	
MD.HRM.PAY070.PUNCH4				1		MD.HRM.PAY070.PUNCH5				1	
MD.HRM.PAY070.PUNCH6				1		MD.HRM.PAY070.PUNCH7				1	
MD.HRM.PAY070.PUNCH8				1		MD.HRM.PAY070.PUNCH9				1	
MD.HRM.QEHCNTL				1		MD.HRM.SAM.BONDTP				1	
MD.HRM.SAM.CHGFILE				1		MD.HRM.SAM.CHKNBR				1	
MD.HRM.SAM.CHKOUT				1		MD.HRM.SAM.CHKREV				1	
MD.HRM.SAM.CNTLFIL				1		MD.HRM.SAM.CURLAB				1	

TSIDOFL Report

Description TSIDOFL creates a report listing the datasets offloaded or backed up. To produce this report, see 13-22 in this manual.

Fields

DATASET NAME is the name of the dataset offloaded or backed up.
VER is the dataset's version number.
SERIAL is the volume serial number the dataset came from.
CUU is the drive address the dataset came from.
EXTENTS START is the former extent starting address.
EXTENTS END is the former extent ending address.
EXTENTS SIZE is the size of the extent.
CREATE DATE is the date the dataset was created.
EXPIRE DATE is the date the dataset is scheduled to expire.
NUMBER RECORDS is the number of records in the dataset.
BLOCK SIZE is the block size of the dataset.
RECORD SIZE is the record size.
TYPE is the file type.
FILE SEQ is the dataset's file sequence on the offload tape.

Dataset Offload Report Example

TSIDOFL= 6.1									
TIME: 9.15.09									
D A T A S E T O F F L O A D R E P O R T									
-----EXTENTS-----									
CREATE EXPIRE BLOCK RECORD									
DATE 12/12/1997									
DATASET NAME VER SERIAL CUU START END SIZE DATE DATE SIZE SIZE RECFM									
VERIFY3.OFLONL01.FBA.RECFM.V G=0000002 1 EPV400 400 9383 9458 76 1997346 1997346 1000 1000 VB									
VERIFY3.OFLONL01.FBA.RECFM.U G=0000002 1 EPV400 400 9318 9382 65 1997346 1997346 100 100 U									
VERIFY3.OFLONL01.FBA.RECFM.S G=0000002 1 EPV400 400 9242 9317 76 1997346 1997346 1000 1000 VBS									
VERIFY3.OFLONL01.FBA.RECFM.F G=0000002 1 EPV400 400 9090 9165 76 1997346 1997346 1000 100 FB									
VERIFY3.OFLONL01.CKD.RECFM.V G=0000002 1 CKD001 160 9535 9537 3 1997346 1997346 1234 1234 VB									
VERIFY3.OFLONL01.CKD.RECFM.U G=0000002 1 CKD001 160 9528 9534 7 1997346 1997346 100 100 U									
VERIFY3.OFLONL01.CKD.RECFM.S G=0000002 1 CKD001 160 9524 9527 4 1997346 1997346 1000 1000 VBS									
VERIFY3.OFLONL01.CKD.RECFM.F G=0000002 1 CKD001 160 9522 9523 2 1997346 1997346 1000 100 FB									
OFFLOAD DATA SET NAME = VERIFY3.001.TAPE VOLUME SERIAL = VRFY31									
TSIDOFL= 6.1									
TIME: 9.15.09									
D A T A S E T O F F L O A D R E P O R T									
-----EXTENTS-----									
CREATE EXPIRE BLOCK RECORD									
DATE 12/12/1997									
DATASET NAME VER SERIAL CUU START END SIZE DATE DATE SIZE SIZE RECFM									
VOLUME TRACKS RELEASED TOTAL EXTENTS									
EPV400 293 4									
CKD001 16 4									
* GRAND TOTAL * 309 8									

TSIDONL Report

Description

TSIDONL creates a report listing the datasets that were onloaded. To create this report, see "Onloading and Restoring Datasets."

Fields

DATASET NAME is the name of the dataset onloaded.
 OLD VER is the version number of the dataset before it was onloaded.
 OLD GEN is the generation number of the dataset before it was onloaded.
 NEW VER is the version number of the dataset after it was onloaded.
 NEW GEN is the generation number of the dataset after it was onloaded.
 NUMBER RECORDS is the number of records in the dataset.
 BLOCK SIZE is the block size.
 RECORD SIZE is the record size.
 RECFM is the record format.
 INPUT DATASET NAME is the name of the offload dataset.
 INPUT SERIAL is the volser of the offload tape the dataset was copied from.
 FILE SEQ is the dataset's file sequence on the offload tape.

Dataset Onload Report Example

TSIDONL= 6.1	D A T A S E T O N L O A D R E P O R T										PAGE 1
TIME: 9.18.55											DATE 12/12/1997
	DATA SET NAME	GEN	VER	GEN	BLOCK SIZE	RECORD SIZE	RECFM	DATA SET NAME	SERIAL	FILE SEQ	
	VERIFY3.OFLONL01.FBA.RECFM.V	2	1	3	1000	1000	VB	VERIFY3.001.TAPE	VRFY31	1	
	VERIFY3.OFLONL01.FBA.RECFM.U	2	1	3	100	100	U	VERIFY3.001.TAPE	VRFY31	1	
	VERIFY3.OFLONL01.FBA.RECFM.S	2	1	3	1000	1000	VBS	VERIFY3.001.TAPE	VRFY31	1	
	VERIFY3.OFLONL01.FBA.RECFM.F	2	1	3	1000	100	FB	VERIFY3.001.TAPE	VRFY31	1	
	VERIFY3.OFLONL01.CKD.RECFM.V	2	1	3	1234	1234	VB	VERIFY3.001.TAPE	VRFY31	1	
	VERIFY3.OFLONL01.CKD.RECFM.U	2	1	3	100	100	U	VERIFY3.001.TAPE	VRFY31	1	
	VERIFY3.OFLONL01.CKD.RECFM.S	2	1	3	1000	1000	VBS	VERIFY3.001.TAPE	VRFY31	1	
	VERIFY3.OFLONL01.CKD.RECFM.F	2	1	3	1000	100	FB	VERIFY3.001.TAPE	VRFY31	1	

TSIDPRV Report

Description

TSIDPRV produces the Vaulting Method Report. The fields in this report are:

Fields

DATASET NAME is the name of the dataset.

NO. CYC is the number of cycles to be maintained for the dataset.

DAYS RET is the number of days each version is to be retained.

METH NO. is the vaulting method assigned to the dataset.

VER NO. and VAULT NAME represent the heart of the vaulting method report. They correlate each version number with its designated vault location.

TSIDPRV Report Example

TIME 14 09 28	* * VAULTING METHOD				REPORT * *	DATE 12/04/1997	PAGE 1
DATA SET NAME	NO. CYC	DAYS RET	METH NO.	VER NO.	VAULT NAME		
AR.PROD.NCL.ELME3BKP.GVEXPORT.OFFSITE	30	1	1	1	MAIN LIBRARY		
				2	VAULT 1		
				3	VAULT 2		
AR.PROD.NCL.EREXTR.EL860.OFFSITE	30	1	1	1	MAIN LIBRARY		
				2	VAULT 1		
				3	VAULT 2		
AR.PROD.NCL.ERPND.B.EL860.OFFSITE	30	1	1	1	MAIN LIBRARY		
				2	VAULT 1		
				3	VAULT 2		
AR.PROD.NCL.ERP.YAJ.EL860.OFFSITE	30	1	1	1	MAIN LIBRARY		
				2	VAULT 1		
				3	VAULT 2		
BUDGET.SQLBKUP	3		1	1	MAIN LIBRARY		
				2	VAULT 1		
				3	VAULT 2		
CL.PROD.LGX.ELEXTR.CLEXTBKP.CLDLY	7	7	2	1	VAULT 1		
				2	VAULT 2		
				3	VAULT 3		
				4	MAIN LIBRARY		
CL.PROD.LGX.ELEXTR.CLEXT.EOM.OFFSITE	30	1	1	1	MAIN LIBRARY		
				2	VAULT 1		
				3	VAULT 2		
CL.PROD.LGX.ELHAF.CLHAF.CPY.MONTHEND	2	45	3	1	MAIN LIBRARY		
				2	MAIN LIBRARY		
				3	VAULT 3		
				4	VAULT 1		
				5	MAIN LIBRARY		
CL.PROD.LGX.EL000BKP.CLEOMREN.OFFSITE	30	1	1	1	MAIN LIBRARY		
				2	VAULT 1		
				3	VAULT 2		
CONSTR.BACKUP	3		1	1	MAIN LIBRARY		
				2	VAULT 1		
				3	VAULT 2		
CONSTR.JOURNAL.BK	3		1	1	MAIN LIBRARY		
				2	VAULT 1		
				3	VAULT 2		
CR.PROD.LGX.EL000BKP.ELME1REN.OFFSITE	30	1	1	1	MAIN LIBRARY		
				2	VAULT 1		
				3	VAULT 2		
CR.PROD.LGX.EL000BKP.GVEXPORT.OFFSITE	30	1	1	1	MAIN LIBRARY		
				2	VAULT 1		
				3	VAULT 2		

TSIDSDS Report

Description

TSIDSDS produces a summary report correlating dataset versions and tape volume serial numbers. It is listed in volume serial number order. For information on creating TSIDSDS reports, see 12-25 in this manual.

Fields

There is at least one entry for each version of a dataset. If a version spans more than one volume, there is an entry for each volume.

SER NO is the volume serial number.

DATASET NAME is the name of the dataset the referenced volume contains.

TSIDSDS Report Example

TSIDSDS= 6.01.00		D A T A S E T M A N A G E M E N T S Y S T E M		PAGE 1	
TIME 14 07 58				DATE 12/04/1997	
SER NO	--DATA SET NAME--	SER NO	--DATA SET NAME--		
B12017S	EPIC.DISK01.BKUP	000011	SQL.BKW.F45		
B12110S	EPIC.DISK01.BKUP	000011	SQL.BKW.F46		
B12183S	EPIC.DISK01.BKUP	000011	SQL.BKW.F47		
B12437S	EPIC.DISK01.BKUP	000011	SQL.BKW.F48		
B12500S	EPIC.DISK01.BKUP	000011	SQL.BKW.F50		
B12547S	EPIC.DISK01.BKUP	000015	SQL.BKW.F01		
B12862S	EPIC.DISK01.BKUP	000015	SQL.BKW.F02		
B12873S	EPIC.DISK01.BKUP	000015	SQL.BKW.F03		
HEW001	IDCAMS.BACKUP	000015	SQL.BKW.F04		
JA0001	IDCAMS.BACKUP	000015	SQL.BKW.F05		
T12994	T.ZEKE.AUDIT.BKUP	000015	SQL.BKW.F06		
T13014S	T.ZEKE.AUDIT.BKUP	000015	SQL.BKW.F07		
000001	ELECTRIC.METER.BK	000015	SQL.BKW.F08		
000004	SQL.BKC.F01	000015	SQL.BKW.F09		
000004	SQL.BKC.F02	000015	SQL.BKW.F10		
000004	SQL.BKC.F03	000015	SQL.BKW.F11		
000004	SQL.BKC.F04	000015	SQL.BKW.F12		
000004	SQL.BKC.F05	000015	SQL.BKW.F13		
000004	SQL.BKC.F06	000015	SQL.BKW.F14		
000004	SQL.BKC.F07	000015	SQL.BKW.F15		
000004	SQL.BKC.F08	000015	SQL.BKW.F16		
000004	SQL.BKC.F09	000015	SQL.BKW.F17		
000004	SQL.BKC.F10	000015	SQL.BKW.F18		
000004	SQL.BKC.F11	000015	SQL.BKW.F19		
000004	SQL.BKC.F12	000015	SQL.BKW.F20		
000004	SQL.BKC.F13	000015	SQL.BKW.F21		
000004	SQL.BKC.F14	000015	SQL.BKW.F22		
000004	SQL.BKC.F15	000015	SQL.BKW.F23		
000004	SQL.BKC.F16	000015	SQL.BKW.F24		
000004	SQL.BKC.F17	000015	SQL.BKW.F26		
000004	SQL.BKC.F20	000015	SQL.BKW.F27		
000006	JOB.HIST.MONTH	000015	SQL.BKW.F28		
000007	PMTHIST.BKUP	000015	SQL.BKW.F29		
000009	SQL.BKC.F01	000015	SQL.BKW.F30		
000009	SQL.BKC.F02	000015	SQL.BKW.F31		
000009	SQL.BKC.F03	000015	SQL.BKW.F32		
000009	SQL.BKC.F04	000015	SQL.BKW.F33		
000009	SQL.BKC.F05	000015	SQL.BKW.F34		
000009	SQL.BKC.F06	000015	SQL.BKW.F35		
000009	SQL.BKC.F07	000015	SQL.BKW.F36		
000009	SQL.BKC.F08	000015	SQL.BKW.F37		
000009	SQL.BKC.F09	000015	SQL.BKW.F38		
000009	SQL.BKC.F10	000015	SQL.BKW.F39		
000009	SQL.BKC.F11	000015	SQL.BKW.F40		
000009	SQL.BKC.F12	000015	SQL.BKW.F41		
000009	SQL.BKC.F13	000015	SQL.BKW.F42		
000009	SQL.BKC.F14	000015	SQL.BKW.F43		
000009	SQL.BKC.F15	000015	SQL.BKW.F44		
000009	SQL.BKC.F16	000015	SQL.BKW.F45		
000009	SQL.BKC.F17	000015	SQL.BKW.F46		
000009	SQL.BKC.F20	000015	SQL.BKW.F47		

TSIDTAC Report

Description

TSIDTAC creates the Date Time Activity Report, which lists job accounting information by CPU and by date and time. It is a cross reference to the Job Time Activity report.

Fields

CPU NO. is the CPU being reported.

RUN DATE is the date the job ran.

JOB NAME is the name of the job which initiated the activity.

PHASE NAME is the name of the program being reported.

PART ID is the partition the job ran in.

START TIME is the time (in hh.mm.ss format) the phase started.

END TIME is the time the phase completed.

DURATION TIME is the length of time the phase ran.

IDLE TIME is the amount of time the CPU was idle while the phase ran. (Idle time uses standard IBM definitions.)

CPU TIME is the amount of CPU time the program used.

OVERHEAD is the program overhead, as defined by IBM.

TOTAL I/O is the amount of I/O issued by the program to that device.

In addition, grand totals for DURATION TIME, IDLE TIME, CPU TIME, OVERHEAD, and TOTAL I/O are listed at the end of the report.

Date Time Activity Report Example

C.P.U. NO. 111111		RUN DATE 12/04/1997		* J O B A C C O U N T I N G S Y S T E M				TIME 13.20.14		TSIDTAC= 6.01.00	
RUN DATE	JOB NAME	PHASE NAME	PART ID	START TIME	END TIME	DURATION TIME	IDLE TIME	CPU TIME	OVER HEAD	PAGE 1	TOTAL I/O
12/02/1997	BGINIT	IESWAIT	BG	09.29.40	09.30.10	00.00.30	00.00.00	00.00.3	00.00.2		238
12/02/1997	DUMMY		BG	09.30.15	09.30.17	00.00.02	00.00.05	00.00.1	00.00.1		44
12/02/1997	VERIFYJ1	VERIFYJO	BG	09.56.19	09.56.39	00.00.20	00.26.02	00.00.7	00.00.5		707
12/02/1997	VERIFYJ1	VERIFYJO	BG	09.57.47	09.58.05	00.00.18	00.01.08	00.00.7	00.00.5		718
12/02/1997	VERIFYJ1	VERIFYJO	BG	10.01.52	10.02.16	00.00.24	00.03.47	00.00.7	00.00.5		637
12/02/1997	VERIFYJ1	VERIFYJO	BG	10.03.40	10.04.05	00.00.25	00.01.24	00.00.7	00.00.5		637
12/02/1997	VERIFYJ1	JCOPY	BG	10.08.53	10.09.02	00.00.09	00.04.48	00.00.7	00.00.3		373
12/02/1997	VERIFYJ1	VERIFYJI	BG	10.09.02	10.09.19	00.00.17	00.00.00	00.00.6	00.00.5		613
12/02/1997	VERIFYJ1	JCOPY	BG	10.10.10	10.10.16	00.00.06	00.00.51	00.00.7	00.00.3		364
12/02/1997	VERIFYJ1	VERIFYJI	BG	10.10.17	10.10.31	00.00.14	00.00.01	00.00.6	00.00.5		613
12/02/1997	VERIFYJ1	JCOPY	BG	10.14.20	10.14.33	00.00.13	00.03.49	00.00.6	00.00.3		298
12/02/1997	VERIFYJ1	VERIFYJI	BG	10.14.33	10.14.55	00.00.22	00.00.00	00.00.5	00.00.4		570
12/02/1997	VERIFYJ1	JCOPY	BG	10.15.59	10.16.10	00.00.11	00.01.04	00.00.6	00.00.3		296
12/02/1997	VERIFYJ1	VERIFYJI	BG	10.16.11	10.16.25	00.00.14	00.00.01	00.00.5	00.00.5		568
						00.03.45*	00.43.00*	00.07.3*	00.04.7*		6676*
12/02/1997	TSIDRFS	TSIDRFS	F4	09.28.33	09.30.03	00.01.30	00.00.00	00.00.5	00.00.3		465
12/02/1997	JAY	JCOPY	F4	09.44.59	09.45.22	00.00.23	00.14.56	00.01.1	00.00.7		705
12/02/1997	JAY	VERIFYJI	F4	09.45.22	09.45.39	00.00.17	00.00.00	00.00.5	00.00.5		524
12/02/1997	JAY	JCOPY	F4	09.45.39	09.45.45	00.00.06	00.00.00	00.00.8	00.00.3		336
12/02/1997	JAY	TSIDLVT	F4	09.45.46	09.46.15	00.00.29	00.00.01	00.01.7	00.00.7		696
12/02/1997	JAY	JCOPY	F4	09.46.15	09.46.31	00.00.16	00.00.00	00.01.1	00.00.4		1448
12/02/1997	JAY	JCOPY	F4	09.46.32	09.46.38	00.00.06	00.00.01	00.00.6	00.00.3		245
12/02/1997	JAY	JCOPY	F4	09.46.38	09.46.53	00.00.15	00.00.00	00.00.6	00.00.4		357
12/02/1997	JAY	VERIFYJI	F4	09.46.54	09.47.07	00.00.13	00.00.01	00.00.5	00.00.5		521
12/02/1997	JAY	JCOPY	F4	09.47.07	09.47.14	00.00.07	00.00.00	00.00.7	00.00.3		336
12/02/1997	JAY	TSIDLVT	F4	09.47.14	09.47.33	00.00.19	00.00.00	00.01.7	00.00.7		684
12/02/1997	JAY	JCOPY	F4	09.47.33	09.47.47	00.00.14	00.00.00	00.01.1	00.00.4		1457
12/02/1997	JAY	JCOPY	F4	09.47.48	09.47.56	00.00.08	00.00.01	00.00.6	00.00.3		400
12/02/1997	JAY	JCOPY	F4	09.50.11	09.50.31	00.00.20	00.02.15	00.01.0	00.00.6		635
12/02/1997	JAY	VERIFYJI	F4	09.50.32	09.50.44	00.00.12	00.00.01	00.00.5	00.00.5		485
12/02/1997	JAY	TSIDLVT	F4	09.50.45	09.51.02	00.00.17	00.00.01	00.01.7	00.00.7		684
12/02/1997	JAY	JCOPY	F4	09.51.03	09.51.07	00.00.04	00.00.01	00.01.0	00.00.4		1446
12/02/1997	JAY	JCOPY	F4	09.51.08	09.51.11	00.00.03	00.00.01	00.00.5	00.00.3		248
12/02/1997	JAY	JCOPY	F4	09.51.11	09.51.17	00.00.06	00.00.00	00.00.6	00.00.3		284
12/02/1997	JAY	VERIFYJI	F4	09.51.17	09.51.31	00.00.14	00.00.00	00.00.5	00.00.5		487
12/02/1997	JAY	TSIDLVT	F4	09.51.32	09.51.54	00.00.22	00.00.01	00.01.7	00.00.7		695
12/02/1997	JAY	JCOPY	F4	09.51.54	09.52.06	00.00.12	00.00.00	00.01.1	00.00.4		1446
12/02/1997	JAY	JCOPY	F4	09.52.07	09.52.15	00.00.08	00.00.01	00.00.6	00.00.3		404
12/02/1997	JAY	JCOPY	F4	09.56.03	09.56.46	00.00.43	00.03.48	00.00.8	00.00.5		595
12/02/1997	JAY	JCOPY	F4	09.56.46	09.56.52	00.00.06	00.00.00	00.00.7	00.00.2		336
12/02/1997	JAY	TSIDLVT	F4	09.56.53	09.57.16	00.00.23	00.00.01	00.01.7	00.00.7		684
12/02/1997	JAY	JCOPY	F4	09.57.16	09.57.35	00.00.19	00.00.00	00.01.1	00.00.4		1457
12/02/1997	JAY	JCOPY	F4	09.57.35	09.57.43	00.00.08	00.00.00	00.00.6	00.00.3		244
12/02/1997	JAY	JCOPY	F4	09.57.44	09.58.10	00.00.26	00.00.01	00.00.4	00.00.2		251
12/02/1997	JAY	JCOPY	F4	09.58.10	09.58.15	00.00.05	00.00.00	00.00.7	00.00.2		336
12/02/1997	JAY	TSIDLVT	F4	09.58.16	09.58.34	00.00.18	00.00.01	00.01.7	00.00.8		686
12/02/1997	JAY	JCOPY	F4	09.58.35	09.58.48	00.00.13	00.00.01	00.01.1	00.00.4		1446
12/02/1997	JAY	JCOPY	F4	09.58.48	09.58.55	00.00.07	00.00.00	00.00.6	00.00.3		389
12/02/1997	JAY	JCOPY	F4	10.01.37	10.02.23	00.00.46	00.02.42	00.00.9	00.00.5		613
12/02/1997	JAY	TSIDLVT	F4	10.02.24	10.02.58	00.00.34	00.00.01	00.01.7	00.00.7		684
12/02/1997	JAY	JCOPY	F4	10.02.58	10.03.23	00.00.25	00.00.00	00.01.1	00.00.4		1457
12/02/1997	JAY	JCOPY	F4	10.03.24	10.03.34	00.00.10	00.00.01	00.00.5	00.00.3		242
12/02/1997	JAY	JCOPY	F4	10.03.34	10.04.11	00.00.37	00.00.00	00.00.4	00.00.2		269
12/02/1997	JAY	TSIDLVT	F4	10.04.11	10.04.29	00.00.18	00.00.00	00.01.7	00.00.7		684
12/02/1997	JAY	JCOPY	F4	10.04.30	10.04.42	00.00.12	00.00.01	00.01.1	00.00.4		1455
12/02/1997	JAY	JCOPY	F4	10.04.42	10.04.50	00.00.08	00.00.00	00.00.6	00.00.3		389
12/02/1997	JAY	JCOPY	F4	10.08.41	10.09.22	00.00.41	00.03.51	00.00.9	00.00.5		600
12/02/1997	JAY	JCOPY	F4	10.09.23	10.09.27	00.00.04	00.00.01	00.00.7	00.00.2		336
12/02/1997	JAY	TSIDLVT	F4	10.09.28	10.09.45	00.00.17	00.00.01	00.01.7	00.00.7		684
12/02/1997	JAY	JCOPY	F4	10.09.45	10.09.57	00.00.12	00.00.00	00.01.1	00.00.4		1446
12/02/1997	JAY	JCOPY	F4	10.09.58	10.10.06	00.00.08	00.00.01	00.00.5	00.00.3		244
12/02/1997	JAY	JCOPY	F4	10.10.06	10.10.35	00.00.29	00.00.00	00.00.4	00.00.2		256
12/02/1997	JAY	JCOPY	F4	10.10.35	10.10.37	00.00.02	00.00.00	00.00.7	00.00.2		336
12/02/1997	JAY	TSIDLVT	F4	10.10.38	10.10.48	00.00.10	00.00.01	00.01.7	00.00.7		693
12/02/1997	JAY	JCOPY	F4	10.10.48	10.10.56	00.00.08	00.00.00	00.01.1	00.00.4		1446
12/02/1997	JAY	JCOPY	F4	10.10.56	10.11.09	00.00.13	00.00.00	00.00.6	00.00.3		398
12/02/1997	JAY	JCOPY	F4	10.14.05	10.15.01	00.00.56	00.02.56	00.00.9	00.00.5		619
12/02/1997	JAY	TSIDLVT	F4	10.15.01	10.15.23	00.00.22	00.00.00	00.01.7	00.00.7		693
12/02/1997	JAY	JCOPY	F4	10.15.24	10.15.38	00.00.14	00.00.01	00.01.1	00.00.4		1446
12/02/1997	JAY	JCOPY	F4	10.15.38	10.15.51	00.00.13	00.00.00	00.00.5	00.00.3		251
12/02/1997	JAY	JCOPY	F4	10.15.52	10.16.30	00.00.38	00.00.01	00.00.4	00.00.2		275
12/02/1997	JAY	TSIDLVT	F4	10.16.30	10.16.58	00.00.28	00.00.00	00.01.9	00.00.7		693
12/02/1997	JAY	JCOPY	F4	10.16.58	10.17.15	00.00.17	00.00.00	00.01.1	00.00.4		1446
12/02/1997	JAY	JCOPY	F4	10.17.16	10.17.24	00.00.08	00.00.01	00.00.6	00.00.3		398
12/02/1997	STEPSEQ	LIBR	F4	10.21.52	10.23.15	00.01.23	00.04.28	00.10.6	00.02.8		16714
12/02/1997	STEPSEQ	JCOPY	F4	10.23.16	10.26.12	00.02.56	00.00.01	00.34.0	00.08.9		17113
12/02/1997	STEPSEQ	LIBR	F4	10.26.12	10.26.18	00.00.06	00.00.00	00.00.8	00.00.3		696
12/02/1997	STEPSEQ	JCOPY	F4	10.26.19	10.26.25	00.00.06	00.00.01	00.00.4	00.00.2		238
12/02/1997	VERIFYK	JCOPY	F4	11.09.31	11.09.40	00.00.09	00.43.06	00.00.9	00.00.5		630

TSIDUTL Reports

Display ALL, Dataset Name Sequence	<p>The Display ALL, SEQ=DSN report gives you a detailed listing of all datasets in dataset name sequence. Each dataset is reported at two levels: BIM-EPIC dataset definition (EDD) and version information.</p>
BIM-EPIC Dataset Definition	<p>The first two lines for each dataset (beginning with the name) provide the dataset definition. The field titles are listed on these two lines.</p> <p>SID is the system ID. UID is the user ID. RET is the number of days each version is to be retained. CYC is the number of versions (cycles) to be maintained. RBLK is the default block size. REC is the actual record size of the last version. BLK is the actual block size of the last version. VLT is the number of the vault method assigned to the dataset. MODE is the default tape density. REV indicates whether this is a revolving tape dataset. UNLBL indicates whether this is a standard label tape dataset. ENCODE=NO indicates whether tape encryption is activated. OWN is the owner ID. TPOOL is the tape pool ID.</p> <p>In addition to these fields, you may also see the following information in this area of the report:</p> <p>Any Comment associated with the dataset appears at the end of the first line. WORK (located between the RET and CYC fields) indicates the dataset is a non-generation dataset.</p>
Version Information	<p>The field titles listed at the top of each page refer to data recorded at the version level. The fields listed are:</p> <p>SERIAL NO is the tape or disk volume serial number. VER. NO is the relative version number of the version. There may also be four status codes listed in this column. An "S" means the version is scratched. An "O" means the version is in open status. A "C" means the version is conditionally cataloged. An "M" means the version is the master dataset of an MCAT group. REEL/EXT is the reel or extent sequence number. CREATION JOB NAME is the name of the job that created this version. CREATION DATE is the date the version was created. EXPIRE DATE is the projected expiration date of the version. TIME is the time (HH.MM) the version was created. TP is the dataset's tape pool. PART is the partition in which the version was created. PHYS. UNIT is the address of the device the version was created on. BLOCK COUNT is the number of blocks read the last time the version was accessed.</p>

LAST ACCESS JB NAME is the name of the last job that accessed this version.

LAST ACCESS DATE is the date the version was last accessed.

COUNT USE is the number of times the tape has been opened since it has contained this version.

CLEAN DATE is the last clean date of the tape. If the dataset is a disk dataset, the display appears in the following format: EXT=starting address/allocation.actual usage

TOTAL USE is the total number of times the tape was opened since the last cleaning.

VLT is the current vault location of this tape.

SLT is the current slot location of this tape.

TOTAL ACTIVE is the total number of active tapes on the TSIDUTL report.

TOTAL SCRATCH is the total number of scratch tapes on the TSIDUTL report.

A-28 User Reference

Volume Serial Number and Vault Sequences

Description

This report lists tape datasets. Each tape volser is listed along with data for the version the tape contains. Dataset definitions are not listed. The first field listed is the version's dataset name. Refer to "Version Data" under "Display ALL, Dataset Name Sequence" for an explanation of the other fields contained in this report.

Example

D A T A S E T M A N A G E M E N T S Y S T E M															PAGE 11		
DSPLY ALL,SEQ=SER															DATE 12/04/1997		
SERIAL	VER.	REEL	---	CREATION	----	EXPIRE				BLOCK	---	LAST	ACCESS---	USE	CLEAN	USE	
NO	NO	/EXT	JB	NAME	DATE	DATE	TIME	TP	PT	CUU	COUNT	JB	NAME	DATE	COUNT	DATE	TOTAL VLT SLT
T.MD.HRM.SAM.RPTREC	DUMMY8	1	1	PAYROL10	05/07/1996	05/07/1996	10.46	BG	1AB	0	EPICONLD	10/18/1996	X=0019644/00200.00003				
T.NAME CI BACKUP	DUMMY8	1	1	DCDMAINT	10/02/1996	10/02/1996	11.59	BG	1AB	0	TSIBACK	08/01/1997	X=0019924/00024.00002				
T.HPL.WORK.FILE	DUMMY8	1	1	HPLVSAMP	10/11/1996	10/11/1996	15.58	BG	1AB	0	EPICONLD	10/18/1996	X=0019720/00100.00002				
T.MD.HRM.SAM.SUP050T	DUMMY8	1	1	PAYROL10	02/21/1996	02/21/1996	15.17	BG	1AB	0	EPICONLD	10/18/1996	X=0019609/00015.00002				
T.OLDNOVER.FILE	DUMMY8	1	1	OCWOPNBL	04/05/1995	04/05/1995	16.00	BG	1AB	0	EPICONLD	10/18/1996	X=0019605/00012.00002				
T.PROG.WORK.FILE3	DUMMY8 O	1	1	HRBDESIG	08/04/1997	08/05/1997	08.06	C1	1AB	0	HRBDESIG	08/04/1997	X=0020793/00050.00050				
T.MD.HRM.SAM.PERWTRN	DUMMY8	1	1	PAYROL10	05/07/1996	05/07/1996	10.30	BG	1AB	0	TSIDONL	11/07/1996	X=0019734/00050.00008				
T.PROG.WORK.FILE3	DUMMY8 O	1	1	HRBDESIG	08/04/1997	08/05/1997	09.07	C2	1AB	0	HRBDESIG	08/04/1997	X=0020457/00050.00050				
T.MD.HRM.SAM.PER171T	DUMMY8	1	1	PAYROL10	03/28/1996	03/28/1996	11.22	BG	1AB	0	EPICONLD	10/18/1996	X=0019611/00015.00002				
T.MD.HRM.PAY070.PUNCH10	DUMMY8	1	1	PAMSTR11	07/09/1997	07/09/1997	10.28	C2	1AB	0	PAMSTR11	07/09/1997	X=0023535/00015.00002				
T.PROJECT.FL	DUMMY8	1	1	DCXHILOW	01/24/1996	01/24/1996	10.24	BG	1AB	0	EPICONLD	10/18/1996	X=0019642/00100.00002				
T.MD.HRM.SAM.PERWORK	DUMMY8	1	1	PAYROL10	05/07/1996	05/07/1996	10.31	BG	1AB	0	EPICONLD	10/18/1996	X=0019613/00015.00002				
T.ELMET.INVENTORY.FILE	DUMMY8	1	1	KLEPIC1	06/02/1994	06/02/1994	10.39	BG	1AB	0	TSIBACK	08/01/1997	X=0020113/00020.00004				
T.JOB.HIST.ACCUM	DUMMY8	2	1	TSIJOBAT	07/31/1997	07/31/1997	06.11	F2	1AB	0	TSIJOBAT	08/01/1997	X=0025648/00100.00100				
T.MD.HRM.SAM.PERTRAN	DUMMY8	1	1	PAYROL10	05/07/1996	05/07/1996	10.31	BG	1AB	0	EPICONLD	10/18/1996	X=0019615/00015.00002				
T.JOB.HIST.ACCUM	DUMMY8	2	3	TSIJOBAT	07/31/1997	07/31/1997	06.11	F2	1AB	0	TSIJOBAT	08/01/1997	X=0023537/00025.00017				
T.TMON.DAILY.D	DUMMY8	1	7	CICDTMON	08/04/1997	08/04/1997	13.36	F2	1AB	0	CICDTMON	08/04/1997	X=0024304/00015.00015				
T.TMON.DAILY.D	DUMMY8	1	6	CICDTMON	08/04/1997	08/04/1997	13.36	F2	1AB	0	CICDTMON	08/04/1997	X=0024289/00015.00015				
T.MD.HRM.SAM.CHKNBRS	DUMMY8	1	1	PAYROL10	05/07/1996	05/07/1996	10.34	BG	1AB	0	TSIDONL	11/07/1996	X=0019635/00005.00002				
T.TMON.DAILY.D	DUMMY8	1	3	CICDTMON	08/04/1997	08/04/1997	13.35	F2	1AB	0	CICDTMON	08/04/1997	X=0020518/00015.00015				
T.MD.HRM.SAM.CNTLFIL	DUMMY8	1	1	PAYROL10	05/07/1996	05/07/1996	10.28	BG	1AB	0	EPICONLD	10/18/1996	X=0019682/00005.00002				
T.FINNODE.562	DUMMY8	1	1	MF22990	07/07/1997	07/07/1997	12.06	C1	1AB	0	TSIBACK	08/01/1997	X=0023646/00012.00012				
T.MD.HRM.SAM.CHKOUT	DUMMY8	1	1	PAYROL10	05/07/1996	05/07/1996	10.45	BG	1AB	0	EPICONLD	10/18/1996	X=0019684/00200.00003				
T.TMON.DAILY.D	DUMMY8	1	4	CICDTMON	08/04/1997	08/04/1997	13.35	F2	1AB	0	CICDTMON	08/04/1997	X=0020762/00015.00015				
T.TMON.DAILY.D	DUMMY8	1	1	CICDTMON	08/04/1997	08/04/1997	13.34	F2	1AB	0	CICDTMON	08/04/1997	X=0024565/00030.00030				
T.MD.HRM.SAM.DATACTL	DUMMY8	1	1	PAYROL10	05/07/1996	05/07/1996	10.28	BG	1AB	0	TSIDONL	11/07/1996	X=0019633/00005.00002				
T.TMON.DAILY.C	DUMMY8 O	1	1	CICCTMON	08/04/1997	08/04/1997	11.17	C2	1AB	0	CICCTMON	08/04/1997	X=0023675/00030.00030				
T.TMON.DAILY.B	DUMMY8 O	2	1	CICBTMON	08/04/1997	08/04/1997	11.08	F5	1AB	0	CICBTMON	08/04/1997	X=0020256/00015.00015				
T.MD.HRM.SAM.ERROR10	DUMMY8	1	1	PAYROL10	05/07/1996	05/07/1996	10.28	BG	1AB	0	EPICONLD	10/18/1996	X=0019680/00050.00002				
T.FINAL ENTREX	DUMMY8	1	1	DCDMAINT	10/02/1996	10/02/1996	12.00	BG	1AB	0	TSIBACK	08/01/1997	X=0020103/00006.00002				
T.MD.HRM.SAM.ERROR20	DUMMY8	1	1	PAYROL10	05/07/1996	05/07/1996	10.33	BG	1AB	0	EPICONLD	10/18/1996	X=0019678/00005.00002				
T.TAX.CODE.FILE	DUMMY8	1	1	DCXCSLST	06/20/1997	06/20/1997	07.47	C1	1AB	0	TSIBACK	08/01/1997	X=0020232/00006.00002				
T.MD.HRM.SAM.ONLCHGS	DUMMY8	1	1	PAYROL10	05/07/1996	05/07/1996	10.28	BG	1AB	0	EPICONLD	10/18/1996	X=0019622/00005.00003				
T.MD.HRM.SAM.MNTRPCH	DUMMY8	1	1	PAYROL10	05/07/1996	05/07/1996	10.33	BG	1AB	0	EPICONLD	10/18/1996	X=0019647/00015.00002				
T.MD.HRM.SAM.CHGFILE	DUMMY8	1	1	PAYROL10	05/07/1996	05/07/1996	10.32	BG	1AB	0	EPICONLD	10/18/1996	X=0019687/00005.00003				
T.TMON.DAILY.A	DUMMY8 O	1	1	CICATMON	08/04/1997	08/04/1997	13.42	F2	1AB	0	CICATMON	08/04/1997	X=0023387/00030.00030				
T.IND.AUDIT.FILE	DUMMY8	1	1	DCXBILLS	07/18/1995	07/18/1995	13.26	BG	1AB	0	TSIBACK	08/01/1997	X=0019926/00005.00003				
T.MD.HRM.SAM.MONITOR	DUMMY8	1	2	PAYROL10	05/07/1996	05/07/1996	10.32	BG	1AB	0	TSIDONL	11/07/1996	X=0019627/00007.00006				
T.TEMP HP HISTORY	DUMMY8	1	1	EPICONLD	10/18/1996	10/18/1996	10.58	BG	1AB	0	TSIBACK	08/01/1997	X=0019800/00006.00004				
T.MD.HRM.SAM.MONITOR	DUMMY8	1	1	PAYROL10	05/07/1996	05/07/1996	10.32	BG	1AB	0	TSIDONL	11/07/1996	X=0019704/00015.00015				
T.MD.HRM.SAM.INTRANS	DUMMY8	1	1	PAYROL10	05/07/1996	05/07/1996	10.29	BG	1AB	0	TSIDONL	11/07/1996	X=0020440/00150.00007				
T.MD.HRM.PAY070.PUNCH6	DUMMY8	1	1	PSMCOPI	05/19/1997	05/19/1997	10.50	C1	1AB	0	HRBEMPRF	05/22/1997	X=0020225/00015.00007				
T.JOB.HIST.ACCUM	DUMMY8	1	1	TSIJOBAT	08/01/1997	08/01/1997	06.11	F2	1AB	0	TSIBACK	08/01/1997	X=0025923/00100.00100				
T.MD.HRM.SAM.LABORT	DUMMY8	1	1	PAYROL10	03/11/1996	03/11/1996	11.21	BG	1AB	0	EPICONLD	10/18/1996	X=0019649/00030.00029				

Expiration Date, Creation Date, and Creation Job Sequences

Each version is listed with all version data. Dataset definitions are not listed. The first field listed is the version's dataset name. Refer to "Version Data" under "Display ALL, Dataset Name Sequence" for an explanation of the other fields contained in this report.

Display Active

This is a detailed report listing the active versions of each dataset. Refer to the "Display ALL, Dataset Name Sequence" for an explanation of the fields contained in this report.

Display NEW Report

This report lists all datasets created since the last time you ran this report. The fields in this report are described under "Display ALL, Dataset Name Sequence".

TSIDUTL NEW Report Example

TSIDUTL= 6.1 TIME 6.45.36															PAGE 1			
D S P L Y N E W , S E Q = D S N															DATE 12/30/1997			
SERIAL VER. REEL --- CREATION ---															USE CLEAN USE			
NO NO /EXT JB NAME DATE DATE TIME TP PT CUU COUNT JB NAME DATE COUNT DATE TOTAL VLT SLT																		
CASH.BUD1.BKUP																		
SID=	UID=	RET=	0	CYC=	1	REBLK=	REC=	8244	BLK=	8244	DLA=	0						
VLT=	0	MODE=F8	REV=NO	UNLBL=NO	ENCODE=NO	OWN=00	TPOOL=B											
000149	1	1	BUMCBKUP	07/07/1997	07/07/1997	09.42	C4	611	1002	BUMCBKUP	07/07/1997	1			14	0	0	
000497	S	1	BUMCBKUP	06/04/1997	06/04/1997	14.18	C2	610	994	BUMCBKUP	06/04/1997	1			4	0	0	
CASH.BUD2.BKUP																		
SID=	UID=	RET=	0	CYC=	1	REBLK=	REC=	8244	BLK=	8244	DLA=	0	NSU FILE OF CASH.BUD1.BKUP					
VLT=	0	MODE=F8	REV=NO	UNLBL=NO	ENCODE=NO	OWN=00	TPOOL=B	SYS000										
F=0000002	000149	1	1	BUMCBKUP	07/07/1997	07/07/1997	09.50	C4	611	26	BUMCBKUP	07/07/1997	1		1	0	0	
CI.PROD.LGX.EL512BKP.GVEXPORT.EL512RST																		
SID=TP	UID=	RET=	1	CYC=	1	REBLK=	REC=	BLK=32767		DLA=	0	LOGIC IDCAMS BACKUP/RESTORE OF						
VLT=	0	MODE=08	REV=NO	UNLBL=NO	ENCODE=NO	OWN=	TPOOL=	SYS000										
011625	1	1	IN512DNH	08/07/1997	08/11/1997	17.37	F3	180	0	TSIDYN	08/18/1997	0			0	0	0	
011967	2	1	IN511MID	08/06/1997	08/10/1997	18.19	F3	180	0	TSIDYN	08/18/1997	0			0	0	0	
011953	3	1	IN511VNB	08/06/1997	08/10/1997	18.14	F3	180	0	TSIDYN	08/18/1997	0			0	0	0	
011938	4	1	IN512DNH	08/06/1997	08/10/1997	17.47	F3	180	0	TSIDYN	08/18/1997	0			0	0	0	
011850	5	1	IN512DNH	08/05/1997	08/09/1997	17.30	F3	180	0	TSIDYN	08/18/1997	0			0	0	0	
011851	6	1	IN511MID	08/05/1997	08/09/1997	17.24	F3	180	0	TSIDYN	08/18/1997	0			0	0	0	
012128	7	1	IN511HEL	08/04/1997	08/08/1997	17.58	F3	180	0	TSIDYN	08/18/1997	0			0	0	0	
012113	8	1	IN512DNH	08/04/1997	08/08/1997	17.52	F3	180	0	TSIDYN	08/18/1997	0			0	0	0	
012109	9	1	IN511FBS	08/04/1997	08/08/1997	17.46	F3	180	0	TSIDYN	08/18/1997	0			0	0	0	
012095	10	1	IN511MTH	08/04/1997	08/08/1997	17.39	F3	180	0	TSIDYN	08/18/1997	0			0	0	0	
012094	11	1	IN511VNB	08/04/1997	08/08/1997	17.33	F3	180	0	TSIDYN	08/18/1997	0			0	0	0	
CI.PROD.NCL.EL511WA.BKS511.ENTRY																		
SID=TP	UID=	RET=	21	CYC=	1	REBLK=	REC=	80	BLK=32000	DLA=	0	LOGIC PROD INPUT DATA TO EL512						
VLT=	0	MODE=08	REV=NO	UNLBL=NO	ENCODE=NO	OWN=	TPOOL=	SYS000										
012107	1	1	IN511FBS	08/04/1997	08/25/1997	17.49	F3	180	0	TSIDYN	08/18/1997	0			0	0		
CI.PROD.NCL.EL511WA.CI003CNV.ENTRY																		
SID=TP	UID=	RET=	1	CYC=	1	REBLK=	REC=	80	BLK=32000	DLA=	0	LOGIC JOURNAL FILE						
VLT=	0	MODE=08	REV=NO	UNLBL=NO	ENCODE=NO	OWN=	TPOOL=	SYS000										
011963	1	1	IN511MID	08/06/1997	08/10/1997	18.22	F3	181	0	TSIDYN	08/18/1997	0			0	0	0	
CI.PROD.NCL.EL511WA.CI005CNV.ENTRY																		
SID=TP	UID=	RET=	21	CYC=	1	REBLK=	REC=	80	BLK=32000	DLA=	0	LOGIC JOURNAL PROD FILE						
VLT=	0	MODE=08	REV=NO	UNLBL=NO	ENCODE=NO	OWN=	TPOOL=	SYS000										
012127	1	1	IN511HEL	08/04/1997	08/25/1997	18.00	F3	180	0	TSIDYN	08/18/1997	0			0	0	0	

**Display PULL
Report**

This report lists only the most current version for each dataset. The fields in this report are described under "Display ALL, Dataset Name Sequence".

Example

D A T A S E T M A N A G E M E N T S Y S T E M														PAGE 1
TSIDUTL= 6.01.00														DATE 12/04/1997
TIME 13.42.37														
DISPLY PULL,SEQ=DSN														
SERIAL VER. REEL ---CREATION----- EXPIRE														
NO NO /EXT JB NAME DATE DATE TIME TP PT CUU COUNT JB NAME DATE DATE TOTAL VLT SLT														
A-R TAPE														
SID= UID= RET= 0 CYC= 2 REBLK=MAX REC= 25 BLK=32750 DLA= 0														
VLT= 0 MODE=F8 REV=NO UNLBL=NO ENCODE=NO OWN=00 TPOOL=B SYS000														
POOL=DISK01 EXT=0000250.0000050 SEC=000 ALR=00000000														
001249 1 1 OCNARBAK 06/30/1997 06/30/1997 20.11 F5 620 342 OCYAUDAR 07/01/1997 2 2 0 0														
ABSENTEE.TEMP														
SID= UID= RET= 0 CYC= 2 WORK REBLK=OPT REC= 1700 BLK=22100 DLA= 0														
POOL=DISK02 EXT=0000060.0000030 SEC=000 ALR=00000000														
DUMMY0 1 1 ABMSETUP 07/08/1997 07/08/1997 11.25 C3 1B7 0 ABMBURA1 07/09/1997 X=0006036/00060.00041														
ACCOUNTS.PAYABLE.FILE														
SID= UID= RET= 0 CYC= 1 REBLK= REC= 80 BLK= 4160 DLA= 0														
POOL=DISK01 EXT=0000060.0000030 SEC=000 ALR=00000000														
G=0000012 DUMMY0 1 1 GAMAPLST 07/16/1997 07/16/1997 18.17 F2 1B7 0 TSIBACK 08/02/1997 X=0005607/00060.00020														
ACCTS.PAYABLE.TWO														
SID= UID= RET= 0 CYC= 1 REBLK= REC= 80 BLK= 4160 DLA= 0														
POOL=DISK01 EXT=0000060.0000030 SEC=000 ALR=00000000														
G=0000012 DUMMY3 1 1 GAMAPLST 07/16/1997 07/16/1997 18.17 F2 1A9 0 TSIBACK 08/02/1997 X=0030740/00060.00020														
ALERT.AUDIT.HIST														
SID= UID= RET= 0 CYC= 3 REBLK= REC=32767 BLK=32767 DLA= 0														
VLT= 0 MODE=F8 REV=NO UNLBL=NO ENCODE=NO OWN=00 TPOOL=B SYS000														
000092 1 1 ALTAUDIT 08/04/1997 08/04/1997 06.48 C1 611 575 ALTAUDIT 08/04/1997 1 17 0 0														
AR CONDENSE FILE														
SID= UID= RET= 0 CYC= 1 REBLK= REC= 5 BLK= 250 DLA= 0														
POOL=DISK01 EXT=0000005.0000002 SEC=000 ALR=00000000														
G=0000132 DUMMY3 1 1 DCDCREAR 08/01/1997 08/01/1997 19.13 C1 1A9 0 TSIBACK 08/02/1997 X=0010300/00005.00002														
AR.PROD.LGX.AR192BKP.GVEXPORT.AR192RST														
SID=TP UID= RET= 2 CYC= 2 REBLK= REC= BLK=32767 DLA= 0 LOGIC IDCAMS BACKUP/RESTORE OF														
VLT= 0 MODE=08 REV=NO UNLBL=NO ENCODE=NO OWN= TPOOL= SYS000														
011609 1 1 AR835 08/07/1997 08/11/1997 16.53 F3 180 0 TSIDYN 08/18/1997 0 0 0 0														
AR.PROD.LGX.EL588BKP.GVEXPORT.EL588RST														
SID=TP UID= RET= 2 CYC= 2 REBLK= REC= BLK=32767 DLA= 0 LOGIC IDCAMS BACKUP/RESTORE OF														
VLT= 0 MODE=08 REV=NO UNLBL=NO ENCODE=NO OWN= TPOOL= SYS000														
011615 1 1 EL588 08/07/1997 08/11/1997 17.31 F3 180 0 TSIDYN 08/18/1997 0 0 0 0														
AR.PROD.NCL.CASHRCT.CASHRBKP.DAILY														
SID=TP UID= RET= 2 CYC= 3 REBLK= REC= 80 BLK= 80 DLA= 0 CASH RECEIPTS DAILY BACKUP														
VLT= 0 MODE=08 REV=NO UNLBL=NO ENCODE=NO OWN= TPOOL= SYS000														
011610 1 1 AR835 08/07/1997 08/11/1997 16.52 F3 180 0 TSIDYN 08/18/1997 0 0 0 0														
AR.PROD.NCL.ELME3BKP.GVEXPORT.ELME3RST														
SID=TP UID= RET= 90 CYC= 3 REBLK= REC= BLK=32767 DLA= 0 LOGIC BACK-UP OFFLINE FILES UP														
VLT= 0 MODE=08 REV=NO UNLBL=NO ENCODE=NO OWN= TPOOL= SYS000														
010523 1 1 ELME3AR 07/19/1997 10/17/1997 18.05 BG 180 0 TSIDYN 08/18/1997 0 0 0 0														
AR.PROD.NCL.ELME3BKP.GVEXPORT.OFFSITE														
SID=TP UID= RET=PERM CYC= 30 REBLK= REC=32763 BLK=32767 DLA= 0 LOGIC BACK-UP OFFLINE FILES CR														
VLT= 1 MODE= REV=NO UNLBL=NO ENCODE=NO OWN= TPOOL= SYS000														
090783 1 1 1996 01/26/1997 06/12/2024 06.52 F5 283 0 TSIDYN 08/18/1997 0 0 1 0														
AR.PROD.NCL.ERBFWD.EL863.MONTHEND														
SID=TP UID= RET= 35 CYC= 1 WORK REBLK= REC= 1000 BLK=23000 DLA= 0														
VLT= 0 MODE=08 REV=NO UNLBL=NO ENCODE=NO OWN= TPOOL= SYS000														
010405 1 1 ELME3AR 07/19/1997 08/23/1997 16.58 BG 181 0 TSIDYN 08/18/1997 0 0 0 0														
AR.PROD.NCL.EREXTR.EL860.MONTHEND														
SID=TP UID= RET= 90 CYC= 3 REBLK= REC= 629 BLK=23273 DLA= 0 LOGIC EL860 ERRORS														
VLT= 0 MODE=08 REV=NO UNLBL=NO ENCODE=NO OWN= TPOOL= SYS000														
012145 1 1 ELME1GET 07/18/1997 10/16/1997 18.20 F3 180 0 TSIDYN 08/18/1997 0 0 0 0														
AR.PROD.NCL.EREXTR.EL860.OFFSITE														
SID=TP UID= RET=PERM CYC= 30 REBLK= REC= 629 BLK=23273 DLA= 0 LOGIC EL860 ERRORS DECEMBER														
VLT= 1 MODE= REV=NO UNLBL=NO ENCODE=NO OWN= TPOOL= SYS000														
090785 1 1 1996 01/26/1997 06/12/2024 07.05 F5 283 0 TSIDYN 08/18/1997 0 0 1 0														
AR.PROD.NCL.ERPND.EL860.MONTHEND														
SID=TP UID= RET= 90 CYC= 3 REBLK= REC= 585 BLK=23400 DLA= 0 LOGIC EL860 PENDING BUSINESS														
VLT= 0 MODE=08 REV=NO UNLBL=NO ENCODE=NO OWN= TPOOL= SYS000														
012149 1 1 ELME1GET 07/18/1997 10/16/1997 18.21 F3 280 0 TSIDYN 08/18/1997 0 0 0 0														
AR.PROD.NCL.ERPND.EL860.OFFSITE														
SID=TP UID= RET=PERM CYC= 30 REBLK= REC= 585 BLK=23400 DLA= 0 LOGIC PENDING BUSINESS DECEMBE														
VLT= 1 MODE= REV=NO UNLBL=NO ENCODE=NO OWN= TPOOL= SYS000														
090786 1 1 1996 01/26/1997 06/12/2024 07.21 F5 283 0 TSIDYN 08/18/1997 0 0 1 0														

This report lists the vault and slot locations of all tapes located in offsite vaults. Refer to "Version Data" under "Display ALL, Dataset Name Sequence" for an explanation of the fields contained in this report.

T	S	I	D	U	T	L	=	6.01.00	T	I	M	E	S	E	T	M	A	N	A	G	E	M	E	N	T	S	Y	S	T	E	M	P	A	G	E	M	E	N	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E	M	T	S	Y	S	T	E	M	P	A	G	E
---	---	---	---	---	---	---	---	---------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Display Scratch Report

Description

This report lists scratch tapes along with data specific to the version the scratch tape contains. Refer to "Version Data" under "Dataset Name Sequence" for an explanation of the fields contained in this report.

Example

TSIDUTL= 6.01.00 TIME 13.42.37		DATA SET MANAGEMENT SYSTEM																		PAGE 1					
		DSPLY SCRATCH																				DATE 12/04/1997			
		SERIAL		VER.	REEL	---CREATION----		EXP		DATE	DATE	TIME	TP	PT	CUU	BLOCK	---LAST ACCESS---		USE	COUNT	CLEAN	USE	TOTAL	VLT	SLT
		NO	NO	NO	/EXT	JB	NAME	DATE	DATE							COUNT	JB	NAME	DATE	COUNT	DATE				
ADFIL.BKUP		001015	S	1		VSNADDFL	07/20/1997	07/23/1997	03.41		C1	610			1508	VSNADDFL	07/20/1997		2			2	0	0	
ALERT.AUDIT.HIST		000298	S	1		ALTAUDIT	05/02/1997	05/02/1997	07.20		C1	620			558	ALTAUDIT	06/03/1997		2			8	0	0	
BUD.ACASH.BKUP		000499	S	1		BUMCBADJ	06/04/1997	06/04/1997	15.08		C1	610			641	BUMCBADJ	06/04/1997		1			3	0	0	
BUDGET.SQLBKUP		000360	S	1		BUMBAKUP	04/30/1997	04/30/1997	07.15		C1	620			169	BUMBAKUP	04/30/1997		1			3	0	0	
CASH.BUD1.BKUP		000497	S	1		BUMCBKUP	06/04/1997	06/04/1997	14.18		C2	610			994	BUMCBKUP	06/04/1997		1			4	0	0	
CONHIST.BACKUP		001014	S	1		VSNCNHST	07/20/1997	07/30/1997	03.33		C2	610			2077	VSNCNHST	07/20/1997		2			2	0	0	
CONSTR.BACKUP		001054	S	1		CAMMASBK	07/22/1997	07/22/1997	11.42		C2	610			0	CAMMASBK	07/22/1997		1			1	0	0	
CONSTR.JOURNAL.BK		000352	S	1		CONSTR	04/30/1997	04/30/1997	07.14		C4	611			214	CONSTR	04/30/1997		1			4	0	0	
ELECTRIC.METER.BK		000429	S	1		MENEMBAK	06/03/1997	06/13/1997	06.47		C1	611			11523	MENEMRST	06/03/1997		2			5	0	0	
EPIC.DBA.WORK1		000724	S	1		BKUPLIBR	06/17/1997	06/24/1997	08.06		BG	611			0	BKUPLIBR	06/17/1997		1			2	0	0	
EPIC.DISK01.BKUP		000301	S	1		TSIBACK	07/02/1997	07/02/1997	02.36		F2	610			3549	TSIBACK	07/02/1997		1			5	0	0	
EPIC.DISK01.BKUP		000258	S	1		TSIBACK	06/28/1997	06/28/1997	02.38		F2	611			5435	TSIBACK	06/28/1997		1			8	0	0	
EPIC.DISK01.BKUP		000281	S	1		TSIBACK	06/24/1997	06/24/1997	02.36		F2	610			5990	TSIBACK	06/24/1997		1			9	0	0	
EPIC.DISK01.BKUP		000240	S	1		TSIBACK	06/21/1997	06/21/1997	05.48		F2	610			3684	TSIBACK	06/21/1997		1			9	0	0	
EPIC.DISK01.BKUP		000712	S	1		TSIBACK	06/17/1997	06/17/1997	04.43		F6	611			5127	TSIBACK	06/17/1997		1			1	0	0	
EPIC.DISK01.BKUP		000661	S	1		TSIBACK	06/10/1997	06/10/1997	02.36		F2	610			5929	TSIBACK	06/10/1997		1			2	0	0	
EPIC.DISK01.BKUP		000609	S	1		TSIBACK	06/07/1997	06/07/1997	02.37		F6	610			4939	TSIBACK	06/07/1997		1			2	0	0	
EPIC.DISK01.BKUP		000566	S	1		TSIBACK	06/06/1997	06/06/1997	02.37		F2	610			4930	TSIBACK	06/06/1997		1			2	0	0	
EPIC.DISK01.BKUP		000525	S	1		NO NAME	06/05/1997	06/05/1997	04.31		F2	610			4910	NO NAME	06/05/1997		1			4	0	0	
EPIC.DISK01.BKUP		000480	S	1		NO NAME	06/04/1997	06/04/1997	04.33		F2	610			4984	NO NAME	06/04/1997		1			3	0	0	
EPIC.DISK01.BKUP		000433	S	1		NO NAME	06/03/1997	06/03/1997	06.49		F2	620			5066	NO NAME	06/03/1997		1			4	0	0	
EPIC.DISK01.BKUP		000454	S	1		NO NAME	05/22/1997	05/22/1997	02.36		F2	610			4500	NO NAME	05/22/1997		1			4	0	0	
EPIC.DISK01.BKUP		000427	S	1		NO NAME	05/21/1997	05/21/1997	08.08		F6	610			5746	NO NAME	05/21/1997		1			4	0	0	
EPIC.DISK01.BKUP		000415	S	1		NO NAME	05/15/1997	05/15/1997	02.36		F2	610			5126	NO NAME	05/15/1997		1			7	0	0	
EPIC.DISK01.BKUP		000422	S	1		NO NAME	05/07/1997	05/07/1997	02.38		F2	611			5800	NO NAME	05/07/1997		1			3	0	0	
EPIC.DISK01.BKUP		000683	S	1		TSIBACK	03/14/1997	03/14/1997	07.04		F2	611			4654	TSIBACK	03/14/1997		1			1	0	0	
EPIC.DISK01.BKUP		B12873	S	1		TSIBACK	02/15/1997	02/15/1997	15.19		F6	803			3483	TSIBACK	02/15/1997		1			1	0	0	
EPIC.DISK01.BKUP		B12110	S	1		TSIBACK	02/14/1997	02/14/1997	05.20		F2	802			3378	TSIBACK	02/14/1997		1			1	0	0	
EPIC.DISK01.BKUP		B12862	S	1		TSIBACK	02/13/1997	02/13/1997	05.21		F6	802			3389	TSIBACK	02/13/1997		1			1	0	0	
EPIC.DISK01.BKUP		B12183	S	1		TSIBACK	02/09/1997	02/09/1997	22.36		BG	801			3194	TSIBACK	02/09/1997		1			1	0	0	
EPIC.DISK01.BKUP		B12017	S	1		TSIBACK	02/02/1997	02/02/1997	00.37		BG	800			3177	TSIBACK	02/02/1997		1			1	0	0	
EPIC.DISK01.BKUP		B12500	S	1		DCMBATCH	01/25/1997	01/25/1997	15.07		W3	000			0	EPICRSTR	01/25/1997		1			1	0	0	
EPIC.DISK01.BKUP		B12547	S	2		DCMBATCH	01/25/1997	01/25/1997	15.07		W3	000			0	DCMBATCH	01/25/1997		0			0	0	0	
EPIC.DISK01.BKUP		B12437	S	3		DCMBATCH	01/25/1997	01/25/1997	15.07		W3	000			0	DCMBATCH	01/25/1997		0			0	0	0	
GAS.METER.BKUP		000223	S	1		MGMBAKUP	07/01/1997	07/01/1997	07.04		C3	610			168	MGMBAKUP	07/01/1997		1			6	0	0	
KKL.TEST.TAPE		000584	S	1		FAV22BK	06/06/1997	06/08/1997	12.05		BG	610			2345	FAV22R	06/06/1997		2			3	0	0	
MBOOKFL.BACKUP		000618	S	1		VSNMBOOK	06/08/1997	06/13/1997	08.06		C1	611			2080	VSNMBOOK	06/08/1997		2			2	0	0	
NOTEFL.BKUP		000221	S	1		CSNNOTBK	08/02/1997	08/03/1997	00.58		C2	611			2225	CSNNOTRS	08/02/1997		2			8	0	0	
NOTEFL.BKUP		000995	S	1		CSNNOTBK	07/19/1997	07/20/1997	01.16		F5	620			2209	CSNNOTRS	07/19/1997		2			2	0	0	
NOTEFL.BKUP		000599	S	1		CSNNOTBK	06/07/1997	06/08/1997	01.22		F5	620			2345	CSNNOTRS	06/07/1997		2			3	0	0	
ODE.P.LIB.ARCHIVE		000987	S	1		ODEBACKP	07/18/1997	07/18/1997	19.16		C4	621			216	ODEBACKP	07/18/1997		1			1	0	0	
ODE.P.LIB.ARCHIVE		000592	S	1		ODEBACKP	06/06/1997	06/06/1997	20.35		C2	611			216	ODEBACKP	06/06/1997		1			2	0	0	
ODE.SAVE1		001092	S	1		ODESAVE	07/23/1997	07/23/1997	19.40		C1	620			804	ODESAVE	07/23/1997		1			1	0	0	
ODE.SAVE1		001063	S	1		ODESAVE	07/22/1997	07/22/1997	19.21		C1	620			835	ODESAVE	07/22/1997		1			1	0	0	
ODE.SAVE1		001032	S	1		ODESAVE	07/21/1997	07/21/1997	20.47		C1	610			809	ODESAVE	07/21/1997		1			1	0	0	
ODE.SAVE1		000986	S	1		ODESAVE	07/18/1997	07/18/1997	19.13		C3	621			793	ODESAVE	07/18/1997		1			1	0	0	
ODE.SAVE1		000955	S	1		ODESAVE	07/17/1997	07/17/1997	18.59		C3	620			736	ODESAVE	07/17/1997		1			1	0	0	
ODE.SAVE1		000921	S	1		ODESAVE	07/16/1997	07/16/1997	19.01		C2	620			728	ODESAVE	07/16/1997		1			1	0	0	
ODE.SAVE1		000888	S	1		ODESAVE	07/15/1997	07/15/1997	19.58		C3	611			741	ODESAVE	07/15/1997		1			1	0	0	
ODE.SAVE1		000852	S	1		ODESAVE	07/14/1997	07/14/1997	19.33		C3	611			724	ODESAVE	07/14/1997		1			1	0	0	
ODE.SAVE1		000260	S	1		ODESAVE	07/07/1997	07/07/1997	20.05		C1	610			815	ODESAVE	07/07/1997		1			12	0	0	
ODE.SAVE1		000440	S	1		ODESAVE	07/02/1997	07/02/1997	19.11		F2	620			787	ODESAVE	07/02/1997		1			3	0	0	
ODE.SAVE1		000697	S	1		ODESAVE	06/16/1997	06/16/1997	21.27		C1	611			642	ODESAVE	06/16/1997		1			1	0	0	
ODE.SAVE1		000646	S	1		ODESAVE	06/09/1997	06/09/1997	20.52		C1	610			646	ODESAVE	06/09/1997		1			2	0	0	
ODE.SAVE1		000591	S	1		ODESAVE	06/06/1997	06/06/1997	20.31		C3	611			630	ODESAVE	06/06/1997		1			2	0	0	
ODE.SAVE1		000551	S	1		ODESAVE	06/05/1997	06/05/1997	20.30		C1	620			595	ODESAVE	06/05/1997		1			1	0	0	
ODE.SAVE1		000510	S	1		ODESAVE	06/04/1997	06/04/1997	20.17		F5	610			585	ODESAVE	06/04/1997		1			3	0	0	
ODE.SAVE1		000389	S	1		ODESAVE	06/02/1997	06/02/1997	21.24		C1	610			726	ODESAVE	06/02/1997		1			5	0	0	
ODE.SAVE1		000671	S	1		ODESAVE																			

Projected Scratch (PSCRATCH) Report

Description

This report lists all tapes that becomes scratch tapes by the specified date. It also reports current DSN Catalog utilization statistics.

The projected date you requested is shown next to the report title PSCRATCH and written in Julian format. For instance PSCRATCH 96001 indicates a projected scratch date of January 1, 1996.

At the end of the report, message EP119 is issued to report on DSN Catalog utilization.

Fields

Name	Description
SERIAL NO	Volume serial number of the tape
FILE NAME	Dataset name of the version written on this tape
CREATION DATE	Date this version was created
EXPIRATION DATE	Date retention criteria may be satisfied for this version
TAPE ERRORS	Number of temporary read/write errors encountered on this tape while it contained this version
TOT-ERR	Total number of errors since the tape was last cleaned
TOT-USED	Total number of times the tape was used since the last cleaning
DT-CLEAN	Date the tape was last cleaned
POOL	
VLT	Current vault location of the tape
SLOT	Current slot location of the tape

PSCRATCH Report Example

TSIDUTL= 6.01.00											
TIME 13.42.37											
PSCRATCH 99095											
DATA SET MANAGEMENT SYSTEM											
PAGE 1											
DATE 12/04/1997											
SERIAL	FILE NAME	CREATION DATE	EXPIRATION DATE	TAPE ERRORS	TOT-ERR	TOT-USED	DT-CLEAN	POOL	VLT	SLOT	
HEW001	IDCAMS.BACKUP	11/14/1997	11/14/1997	0	0	1			0	0	
JA0001	IDCAMS.BACKUP	11/14/1997	11/14/1997	0	0	4			0	0	
000001	ELECTRIC.METER.BK	08/02/1997	08/12/1997	0	0	41			0	0	
000007	PMTHIST.BKUP	08/02/1997	08/09/1997	0	0	42			0	0	
000028	P.SQL.LOG.ARC	08/02/1997	08/09/1997	0	0	28			1	99	
000062	UCAT110A.BKUP	08/03/1997	08/06/1997	0	0	19			0	0	
000063	UCAT111A.BKUP	08/03/1997	08/06/1997	0	0	12			0	0	
000065	UCAT112A.BKUP	08/03/1997	08/06/1997	0	0	12			0	0	
000067	WWSTATS.BKUP.TAPE	07/10/1997	08/09/1997	0	0	9			0	0	
000070	VSAM.BKUP.TEC	08/03/1997	08/05/1997	0	0	14			0	0	
000079	MBOOKFL.BACKUP	08/03/1997	08/08/1997	0	0	12			0	0	
000154	P.SQL.LOG.ARC	07/29/1997	08/05/1997	0	0	16			2	0	

TSIDVLT Reports

Description

TSIDVLT produces two reports listing tapes which should be moved. One is a Picking List which is sorted by the tape's current location. The other is a Distribution List which is sorted by the tapes' new destination. Both reports list the current location of tape reels and their new location. They are designed to assist vault workers in the physical movement of tapes from vault to vault.

To produce these reports, see 13-8 in this manual.

Fields

Name	Description
VAULT INFORMATION & SLOT	Number and name of the vault and the number of the slot from which the tape should be moved
SER-NO	Volume serial number of the tape
VER	Number of the version on the tape
REEL	Reel sequence number of the tape (important for multi-volume versions)
DATASET NAME	Dataset name of the version on the tape
VAULT INFORMATION & SLOT	Number and name of the vault and the number of the slot to which the tape should be moved

Vault Movement Report Example

TSIDVLT VAULT MOVEMENT REPORT					TIME 15 08 50	
DATE 12/11/1997					-----T O-----	
-----F R O M-----					VAULT INFORMATION	
1 VAULT 1	SLOT	SER-NO	VER REEL	DATA SET NAME	0 MAIN LIBRARY	SLOT
1 VAULT 1	112	004207	1 1	T.FCOPY.ESA212	0 MAIN LIBRARY	0
1 VAULT 1	113	004208	1 1	T.FCOPY.ESA213	0 MAIN LIBRARY	0
1 VAULT 1	119	000101	1 1	T.FCOPY.TST701	0 MAIN LIBRARY	0
1 VAULT 1	120	004013	1 1	T.FCOPY.T21RES	0 MAIN LIBRARY	0
1 VAULT 1	121	004014	1 1	T.FCOPY.T21WK1	0 MAIN LIBRARY	0
TSIDVLT VAULT MOVEMENT REPORT					TIME 15 08 50	
DATE 12/11/1997					-----T O-----	
-----F R O M-----					VAULT INFORMATION	
2 VAULT 2	SLOT	SER-NO	VER REEL	DATA SET NAME	1 VAULT 1	SLOT
2 VAULT 2	T12994	2 1	2 1	T.ZEKE.AUDIT.BKUP	1 VAULT 1	110
2 VAULT 2	000118	2 1	2 1	T.KCOPY.TST701	1 VAULT 1	98
2 VAULT 2	000154	2 1	2 1	P.SQL.LOG.ARC	1 VAULT 1	197
2 VAULT 2	000220	2 1	2 1	EPIC.DISK01.BKUP	1 VAULT 1	106
2 VAULT 2	000251	2 1	2 1	CONSTR.JOURNAL.BK	1 VAULT 1	30
2 VAULT 2	000252	2 1	2 1	BUDGET.SQLBKUP	1 VAULT 1	8
2 VAULT 2	000279	2 1	2 1	P.RPFICH.MONTHLY	1 VAULT 1	195
TSIDVLT VAULT MOVEMENT REPORT					TIME 15 08 50	
DATE 12/11/1997					-----F R O M-----	
-----T O-----					VAULT INFORMATION	
1 VAULT 1	SLOT	SER-NO	VER REEL	DATA SET NAME	5	SLOT
1 VAULT 1	1	010185	1 1	PT.A011.CLM.CK.BK	2 VAULT 2	0
1 VAULT 1	2	090651	2 1	AR.PROD.NCL.ELME3BKP.GVEXPORT.OFFSITE	5	0
1 VAULT 1	3	010005	1 1	PT.D089.NAMES	2 VAULT 2	0
1 VAULT 1	4	090653	2 1	AR.PROD.NCL.EREXTR.EL860.OFFSITE	5	0
1 VAULT 1	5	010072	1 1	ST.ADABAS.FILE114.JR114UPD	5	0
TSIDVLT VAULT MOVEMENT REPORT					TIME 15 08 50	
DATE 12/11/1997					-----F R O M-----	
-----T O-----					VAULT INFORMATION	
2 VAULT 2	SLOT	SER-NO	VER REEL	DATA SET NAME	3 VAULT 3	SLOT
2 VAULT 2	1	090529	3 1	AR.PROD.NCL.ELME3BKP.GVEXPORT.OFFSITE	3 VAULT 3	0
2 VAULT 2	2	090531	3 1	AR.PROD.NCL.EREXTR.EL860.OFFSITE	3 VAULT 3	0
2 VAULT 2	3	090532	3 1	AR.PROD.NCL.ERPNDP.EL860.OFFSITE	3 VAULT 3	0
2 VAULT 2	4	090652	3 1	AR.PROD.NCL.ERPYAJ.EL860.OFFSITE	3 VAULT 3	0
2 VAULT 2	5	000432	3 1	BUDGET.SQLBKUP	3 VAULT 3	0

Master Index

BOOK CODES:

(IS) *Installation and System Guide*

(UR) *User Reference*

-- suffix (UR) 1-14, 1-31

-- suffix (UR) 1-14, 1-31

\$

\$\$\$ datasets (UR) 7-31

\$\$BFCMUL (IS) 2-5

\$DEFPROF user profile (UR) 2-12

\$DEFSEC security profile (UR) 2-7, **2-8**

\$EPIC security profile (UR) 2-7, **2-8**

\$EPIC user profile (UR) 2-12

\$JOBEXIT user exit (IS) 2-65

\$SELECT file (IS) 3-15, 3-23

&

&&dsname (UR) 6-23

&dsname (UR) 6-23

&n parameter (UR) 9-12

*

** DSN DELETED ** (UR) 1-35, 2-29, 2-46, 3-22

**** AUTO CATALOGUE **** (UR) 1-19

*** VOLUME.DSN.** (UR) 2-40, 12-9

EPIC.EVENT.DSN (UR) 10-2

=

=== datasets (UR) 7-30

0

000000 volser (UR) 2-40

A

A option (UR) 5-10

Abend processing

DD statements (UR) 6-17

Abend-AID

start before BIM-EPIC (IS) 2-8

Absolute generation

accessing with DLBL option G (UR) 5-10

DD statements (UR) 6-22

Absolute track specification (UR) 6-30

ABSTR parameter (UR) 6-30

ACCEPT configuration option (IS) 5-8

Accepted Tapes Report (UR) 11-6, A-8, A-10

Accessing

absolute generation (UR) 5-10

detail records (UR) 4-11

disk volume (UR) 3-32

DSN Catalog (UR) 4-5

master records (UR) 4-11

Online Manager security (UR) 2-8

relative version (UR) 5-12

sequential and random (UR) 4-5

TOJA (UR) 11-15

VSE label area (UR) 8-21

Accounting statistics (UR) A-3

ACL support

ejecting after MOVE (UR) 3-30

ERD (IS) 2-25

no revolving tapes (UR) 1-12

overriding TSIDSMNT (IS) 2-29

scratching tapes (UR) 3-22

selecting tapes (UR) 2-41

space requirements (IS) 2-7

synchronizing scratch pools (IS) 6-7

ACL support (UR) 2-40

ACLAUTI configuration option (IS) 5-8

- ACLEJECT command (IS) 6-5
- ACLIBM configuration option (IS) 5-8
- ACLeveling (IS) 5-59
- ACLLIB configuration option (IS) 5-8
- ACLLVLOO configuration option (IS) 5-9
- ACLMEM configuration option (IS) 5-9
- ACLOFFL control statement (IS) 2-27
- ACLONL control statement (IS) 2-28
- ACLOPT configuration option (IS) 5-9
- ACLRECS configuration option (IS) 5-9
- ACLSRVRI configuration option (IS) 5-10
- ACLSRVRM configuration option (IS) 5-10
- ACLSRVRO configuration option (IS) 5-10
- ACLSRVRS configuration option (IS) 5-10
- ACLSTK configuration option (IS) 5-10
- ACLTIME configuration option (IS) 5-10
- Activating
 - BIM-EPIC (IS) 5-2
 - IBM ACL support (IS) 5-60
 - Memorex ATL support (IS) 5-63
 - QDAS (IS) 5-52
 - Recorder (IS) 5-45
 - STK ACL support (IS) 5-68
- Active records, number of (UR) 3-11
- Active version (UR) 1-16
- Activity Log Report
 - cross-reference (UR) A-8
 - TSIDARP (UR) 11-3
- Activity Log Report (UR) A-6
- Add a Tape Version panel (UR) 2-34
- ADD command (UR) 1-30, 2-5, 2-34, 2-40, 3-8
- ADD control statement (IS) 2-29
- Adding
 - dummy tape device (IS) 2-61
 - EDDs (UR) 2-23
 - versions (UR) 2-34
- ADDRESS EPIC
 - decoding output from (UR) 4-12
- ADDRESS EPIC (UR) 4-11
- After-hours support xiii, xii
- AGP See Automatic generation purge
- AGP configuration option (UR) 1-26
- ALERT (IS) 2-9, 5-34
- Align on cylinder boundary (UR) 6-30
- ALLMALL configuration option
 - overriding with DLBL option M (UR) 5-11
 - overriding with OPTION (UR) 6-27
- ALLMALL configuration option (IS) 5-10
- ALLMSG configuration option
 - overriding with DLBL option M (UR) 5-11
 - overriding with OPTION (UR) 6-27
- ALLMSG configuration option (IS) 5-11
- ALLMSGSGS subparameter (UR) 6-27
- Allocating
 - LUBs (UR) 1-28
- Allocation Also see ALR and Disk allocation
 - by logical records (UR) 1-14
 - leveling (IS) 2-43
 - LUBS-inhibiting (IS) 2-48
 - on shared volumes (IS) 2-52
- Allocation messages
 - DD statements (UR) 6-26
 - DLBL option M (UR) 5-11
- Allocation size
 - by cylinders (UR) 6-30
 - by logical records (UR) 6-30
 - by tracks (UR) 6-30
 - DD statements (UR) 6-27
 - SPACE parameter (UR) 6-30
- Alpha options on TLBLs (UR) 5-3
- ALR
 - DLBL option R (UR) 5-11
- ALTMSGSGS configuration option (IS) 5-11
- ALX parameter (UR) 6-30
- AMORG DD parameter (UR) 6-12
- AMP parameter (UR) 6-7
- AND condition (UR) 9-9
- APAR information
 - commenting out (IS) 3-25
- APURG (IS) 6-15
- AR facility (UR) 12-5
- ASI procedure
 - adding dummy devices (IS) 2-61
 - modifying for BIM-EPIC (IS) 2-57
 - partition size requirements (IS) 2-63
 - size card (IS) 2-63
 - updating SVA size (IS) 2-63
- ASSGN statements
 - IGNORE (UR) 7-60
 - required with DLBL option I (UR) 5-11
 - TSIDASS (UR) 8-2
 - uncontrolled datasets (UR) 7-7
 - when required (UR) 7-7
- ASSGN statements (UR) 7-7
- Assigning
 - default pools (UR) 1-14
 - SYS number (UR) 1-28
- Assigning vaulting method (UR) A-22
- Assignments
 - holding (UR) 6-24
 - releasing (UR) 6-24
- Assignments (UR) 12-4
- ATL See Memorex ATL support
- ATTACH (UR) 12-4

Attention Routine
 BIM-EPIC Dispatcher (IS) 5-53
 Attention Routine commands (UR) 12-5
 AUTOATT configuration option (IS) 3-2, 5-11
 AUTOATT configuration option (UR) 1-29
 Autoattach
 AUTOATT (IS) 2-94, 5-11
 configuration options (IS) 3-2
 for VM/ESA 370 (IS) 3-41
 for VM/ESA Version 1 Release 1.1 (IS) 3-27
 for VM/ESA Version 1 Release 2 (IS) 3-20
 for VM/ESA Version 2 Release 2 (IS) 3-10
 for VM/HPO (IS) 3-41
 for VM/SP (IS) 3-41
 for VM/XA (IS) 3-35
 installation (IS) 3-1
 overview (IS) 3-2
 via Dynamic Command Definition (IS) 3-4
 Autoattach (UR) 1-29, 12-4
 AUTOCAT configuration option (IS) 5-11
 AUTOCAT configuration option (UR) 1-4, 1-19
 Autodetach restrictions (UR) 1-30
 AUTOLOG configuration option (IS) 5-12
 AUTOLOG configuration option (UR) 1-19
 Automated Cartridge Library
 IBM requirements (IS) 2-5
 Memorex requirements (IS) 2-5
 STK requirements (IS) 2-6
 Automated Cartridge Library (ACL)
 ACCLLeveling (IS) 5-59
 activating and deactivating (IS) 5-63, 5-68
 configuration options (IS) 5-8
 Definition Requirements (IS) 5-54
 IBM ACL (IS) 5-60
 Memorex ATL support (IS) 5-63
 OPTI-TAPE ACL support (IS) 5-73
 STK ACL support (IS) 5-68
 vaulting (UR) 12-17
 Vendors Supported (IS) 5-54
 Automatic Cartridge Library *See* ACL support
 Automatic cataloging
 AUTOCAT (UR) 1-19
 AUTOLOG (UR) 1-19
 DD statements (UR) 6-17, 6-25
 generation datasets (UR) 6-17, 6-22
 SYSPARM (UR) 1-20
 work datasets (UR) 6-17
 Automatic cataloging (UR) 2-40
 Automatic generation purge (UR) **1-26**
 Automatic volume recognition (UR) **1-29**
 Automatic Volume Recognition (UR) 12-4
 AUTOSTP configuration option (IS) 5-12

AUTOSTP configuration option (UR) 1-29
 AUTSTPN configuration option (IS) 5-12
 AUTSTPNO configuration option (IS) 5-12
 AUTSTPT configuration option (IS) 5-12
 AUTSTPTO configuration option (IS) 5-13
 AUX control file (IS) 3-37
 Available drives (UR) 12-4
 Available tapes (UR) 12-4
 AVAULTS (IS) 6-39
 AVAULTS parameter (UR) 12-22
 AVR *See* Automatic volume recognition
 AVR (UR) 1-29

B

B option (UR) 5-10
 Backed-up datasets
 listing (UR) A-20
 BACKUP
 IDCAMS (UR) 7-43
 LIBR (UR) 7-46
 BACKUP command (IS) 6-6
 BACKUP control statement
 RFBKUP (IS) 5-46
 synchronizing DSN Catalog and Recorder (IS) 5-46
 BACKUP control statement (IS) 5-46
 Backup/Offload (UR) 1-31
 BACKUP/RESTORE (IS) 2-11
 Batch catalog management (UR) 1-7
 BCAT control statement (IS) 2-23, 6-21
 BDSPLY command (IS) 6-7
 BIM-BIM-EPIC
 documentation xiv, xiii
 BIM-EPIC
 activating (UR) 1-34
 configuration defaults (IS) 2-94
 configuration options (UR) 1-2
 configuring (IS) 5-2
 dataset control (UR) 1-5
 DD statements (UR) 6-1
 deactivating (IS) 5-44
 deactivating/reactivating (IS) 6-26
 Dispatcher (IS) 5-53
 DLBL enhancements (UR) 5-8
 execution status (UR) 1-34
 first-time installation (IS) 2-4
 JCL enhancements (UR) 1-39
 job cancellation (UR) 1-38
 messages *See* Messages
 request handling (IS) 5-53
 restricting control (UR) 1-5
 special options (IS) 5-31

- special options (UR) 1-22
 - starting (IS) 5-2
 - system files (IS) 2-12
 - TLBL enhancements (UR) 5-2
 - upgrading from previous release (IS) 2-3
 - using JCL (UR) 1-36
 - utility programs (UR) 8-1
 - vaulting (IS) 6-38
 - BIM-EPIC dataset definitions
 - allocation by logical records (UR) 1-14
 - automatic purge (UR) 1-26
 - block size (UR) 1-10
 - CISIZE (UR) 1-14
 - comment (UR) 1-9
 - creating (UR) 2-23
 - cycles retention (UR) 1-11
 - dataset name (UR) 1-9
 - days after last access (UR) 1-11
 - deleting (UR) 2-29, 2-39, 3-22
 - disk pool name (UR) 1-14
 - display format (UR) 2-13
 - displaying (UR) 2-19, 2-22
 - encryption password (UR) 1-9
 - generation dataset (UR) 1-10
 - locking (UR) 2-26
 - managing (UR) 2-21
 - modifying (UR) 3-43
 - number of secondary allocations (UR) 1-14
 - owner ID (UR) 1-12
 - password (UR) 1-9
 - primary extend size (UR) 1-14
 - prime dataset name (UR) 1-12
 - releasing (UR) 2-27
 - removing (UR) 3-33
 - retention (UR) 2-49
 - retention days (UR) 1-10
 - revolving tapes (UR) 1-12
 - secondary extent size (UR) 1-14
 - standard label (UR) 1-12
 - SYS number (UR) 1-10
 - System ID (UR) 1-9
 - tape density (UR) 1-12
 - tape pool ID (UR) 1-12
 - tape size (UR) 1-13
 - TSIDUTL report (UR) A-26
 - unlocking (UR) 2-28
 - updating (UR) 2-25
 - User ID (UR) 1-9
 - vault method number (UR) 1-13
 - BIM-EPIC Dataset Definitions
 - managing (UR) **2-18**
 - BIM-EPIC dataset definitions (UR) A-17
 - BIM-EPIC Resource Dataset (IS) **2-25**
 - BIM-FAQS/ASO
 - accessing Online Manager (UR) 2-3
 - installation (IS) 2-70
 - Online Manager security (UR) 2-7
 - BIM-FAQS/PCS (IS) 2-9
 - BIM-GSS
 - quick installation (IS) 2-2
 - BIM-GSS (UR) 4-11
 - Block size
 - BLKSIZE DD parameter (UR) 6-14
 - MAX (UR) 6-13
 - mutually exclusive with CISIZE (UR) 1-10
 - OPT (UR) 6-13
 - Block size (UR) **1-10**
 - BUFSP DD parameter (UR) 6-12
 - Bypass datasets DD statements (UR) 6-41
 - BYPEXT configuration option (IS) 5-13
 - BYPWRKO parameter (IS) 6-42
-
- ## C
-
- C option (UR) 5-10
 - Caching (IS) 2-17
 - CA-DRIVER
 - start before BIM-EPIC (IS) 2-8
 - CA-DYNAM conversions *See* BIM-EPIC Conversion Guide
 - CA-DYNAM conversions (IS) 5-32
 - CA-Easytrieve (IS) 2-9
 - CA-EPIC for CMS
 - catalog management requirements (UR) 3-5
 - sharing BIM-EPIC catalog (UR) 3-23, 3-27
 - sharing DSN Catalog on VSE (UR) 2-2
 - CA-LIBRARIAN (IS) 2-9
 - Called subroutines (UR) 10-9
 - Cancel codes *See* Job management control language
 - CANMSG configuration option (IS) 5-13
 - CAPACITY command (UR) 2-5, 3-11
 - CA-PANVALET (IS) 2-9
 - Card input (UR) 7-56
 - Cartridge tapes (UR) 1-30
 - CA-SORT (IS) 2-9, 2-11
 - CA-SORT (UR) 7-37
 - CA-Super-EPAT (IS) 5-34
 - CAT control statement (IS) 6-21
 - CAT subparameter (UR) 6-27
 - CATALOG
 - TSIDOFD (UR) 13-20
 - CATALOG command
 - REV (UR) 12-3
 - VLT parameter (UR) 12-19

CATALOG command (UR) 2-24, 2-47, 3-12

Catalog maintenance

SCRATCH (UR) 2-46

Catalog management

ADD (UR) 2-34, 2-40, 3-8

CAPACITY (UR) 3-11

CATALOG (UR) 3-12

CATALOG (UR) 2-24, 2-47

CLEAN (UR) 3-18

CLEAN (UR) 2-45

CLOSE (UR) 3-19

CLOSE (UR) 2-38

CYCLE (UR) 2-51, 3-20

DELETE (UR) 3-22

DELETE (UR) 2-29, 2-39

DEQ (UR) 3-23

DISPLAY (UR) 3-24

DISPLAY (UR) 2-21

ENQ (UR) 3-27

functions (UR) 3-3

LOCK (UR) 3-28

LOCK (UR) 2-26, 2-35

MCATALOG (UR) 3-29

MOVE (UR) 3-30

MOVE (UR) 12-22

online See Online Manager

PASSWORD (UR) 3-7

PURGE (UR) 3-33

PURGE (UR) 2-39, 2-44

RELEASE (UR) 3-35

RELEASE (UR) 2-27, 2-36

RENAME (UR) 3-36

RETAIN (UR) 3-37

RETAIN (UR) 2-49

SCRATCH (UR) 3-39

SCRATCH (UR) 2-46, 2-52

syntax (UR) 3-6

TRANSFER (UR) 3-41

UNLOCK (UR) 3-42

UNLOCK (UR) 2-28, 2-37

UPDATE (UR) 3-43

UPDATE (UR) 2-25, 2-48

vaulting (UR) 12-19

Catalog management (UR) 2-2

Catalog overrides

DD statements (UR) 6-9

Catalog recovery (IS) 4-8

Cataloging datasets

DD statements (UR) 6-9, 6-17, 6-25

CATLG DD parameter (UR) 6-17

CATMOVE_n SAMPJOB (IS) 4-6

CATRT1 configuration option (IS) 5-13

CATRT1 configuration option (UR) 1-4, 1-20

CBL RES option (IS) 2-67

Changing

EDD information (UR) 2-25

vault and slot location (UR) 3-30

version expiration date (UR) 3-37

Changing from

disk to tape (UR) 7-56

tape to disk (UR) 7-56

Check CPU See Job management control language

Check partition See JMCL

Checklists *Also see* Worksheets

initial installation (IS) 1-2

Pre-4.1 Upgrade (IS) 1-4

Checkpoint/restart

TSIDASS (UR) 8-4

Checkpoint/restart (UR) 7-54

Choosing offload datasets (UR) 13-17

CICS

accessing Online Manager (UR) 2-3

event posting entries (IS) 2-78

installation (IS) 2-78

label creation (UR) 12-16

PCT tables (IS) 2-76, 2-77

start-up JCL (IS) 2-86

TOJA (UR) 11-13

TOJA installation (IS) 2-87

transaction activation (IS) 2-94

transaction notes (IS) 2-84

CISIZE

DD parameter (UR) 6-26

mutually exclusive with block size (UR) 1-10

CISIZE (UR) 1-14

CISIZE subparameter (UR) 6-27

CKD devices

allocation on cylinder boundary (UR) 5-10

CKDFACT configuration option (IS) 5-14

CLEAN command (UR) 2-5, 2-43, 2-45, 3-18

CLEAR control statement (IS) 5-47

CLOSE command (UR) 2-5, 2-38, 3-19

CLOSE parameter (UR) 6-24

Close processing

disk files (UR) 1-43

CMDHELP command (UR) 2-16

CMS

accessing Online Manager (UR) 2-3

CMS, sharing catalog with (IS) 2-68

CMSTAPE configuration option (IS) 2-68, 5-14

CNTRL configuration option (IS) 5-15

CNTRL configuration option (UR) 1-5

COBOL

example (UR) 4-7

- Library Management Feature (IS) 2-67
- TSIDACC2 (UR) 4-5
- COBOL copybook (UR) 11-10
- Coding
 - DD statements (UR) 6-2
- Combining DSN Catalogs (IS) 6-70
- Command help (UR) 2-16
- Command syntax xi, x
- Comment (UR) 1-9
- Common work datasets
 - deleting (UR) 1-21
 - retention (UR) 1-21
- Common work datasets (UR) **7-31**
- Compatibility with other software (IS) 2-8
- Compiler work datasets (UR) 7-31, 7-33
- Concatenating
 - not eligible for dataset drop (UR) 6-4
- Concatenating datasets
 - DD statements (UR) 6-3
 - DLBL option E (UR) 5-10
 - multiple opens (UR) 5-12
- Concatenating datasets (UR) 5-7, 7-35, 7-37
- Concatenation
 - datasets example (UR) 7-61
 - DD statements (UR) 6-22
- CONDCAT parameter (UR) 6-17, 7-64
- Condition checking and branching *See* Job management control language
- Conditional cataloging
 - definition (UR) **1-16**
 - DLBL option C (UR) 5-10
- Conditional cataloging (UR) 5-7, 7-64
- Conditionally cataloged
 - CONDCAT parameter (UR) 6-17
- Conditions and IF (UR) 9-9
- Configuration options
 - ACCEPT (IS) 5-8
 - ACLAUTI (IS) 5-8
 - ACLIBM (IS) 5-8
 - ACLLIB (IS) 5-8
 - ACLLVLOO (IS) 5-9
 - ACLMEM (IS) 5-9
 - ACLOPT (IS) 5-9
 - ACLRECS (IS) 5-9
 - ACLSRVRI (IS) 5-10
 - ACLSRVRM (IS) 5-10
 - ACLSRVRO (IS) 5-10
 - ACLSRVRS (IS) 5-10
 - ACLSTK (IS) 5-10
 - ACLTIME (IS) 5-10
 - AGP (UR) 1-26
 - ALLMALL (IS) 5-10
 - ALLMSG (IS) 5-11
 - ALTMSGs (IS) 5-11
 - AUTOATT (IS) 5-11
 - AUTOATT (UR) 1-29
 - Autoattach (IS) 3-2
 - AUTOCAT (IS) 5-11
 - AUTOCAT (UR) 1-4, 1-19
 - AUTOLOG (IS) 5-12
 - AUTOLOG (UR) 1-19
 - AUTOSTP (IS) 5-12
 - AUTOSTP (UR) 1-29
 - AUTSTPN (IS) 5-12
 - AUTSTPNO (IS) 5-12
 - AUTSTPT (IS) 5-12
 - AUTSTPTO (IS) 5-13
 - BYPEXT (IS) 5-13
 - CANMSG (IS) 5-13
 - CATRT1 (IS) 5-13
 - CATRT1 (UR) 1-4, 1-20
 - CKDFACT (IS) 5-14
 - CMSTAPE (IS) 2-68, 5-14
 - CNTRL (IS) 5-15
 - CNTRL (UR) 1-5
 - CPUIDS (IS) 5-15
 - CWDSPOL (IS) 5-15
 - CYCL (IS) 5-15
 - CYCL (UR) 1-20
 - DDERR (IS) 5-16
 - DDGEN (IS) 5-16
 - DDGEN (UR) 6-17, 6-22
 - DEBUG (IS) 5-17
 - defaults (IS) 2-94, 5-4
 - DEFEXT (UR) 1-20
 - DEFPOL (IS) 5-17
 - DEFPOL (UR) 1-14, 1-20
 - DEFTPL (IS) 5-17
 - DEFWPOL (IS) 5-17
 - DISKALL (IS) 5-17
 - DSKLUB (IS) 5-18
 - DSKRET (IS) 5-18
 - DSKRET (UR) 1-24
 - DSNAME (IS) 5-18
 - DSNWARN (IS) 5-18
 - DSNWMSG (IS) 5-18
 - EPICOFF (IS) 5-18
 - ERDFILE (IS) 2-25, 5-19
 - EURODAT (IS) 5-19
 - FBAFACT (IS) 5-19
 - FRAG (IS) 5-19
 - GDI (IS) 5-19
 - GDI (UR) 8-6, 8-9, 8-16
 - GDIUR (IS) 5-20

- JCLAC (IS) 5-20
- JCLLANG (IS) 5-20
- JCLLANG (UR) 9-1
- JCLOPT (IS) 5-20
- JCLOVRDE (IS) 5-20
- JCLOVRDE (UR) 6-9, 6-17
- JOBACCT (IS) 5-20
- JOBACCT (UR) 11-1
- JQNAME (IS) 5-20
- JQWARN (IS) 5-21
- JQWMSG (IS) 5-21
- Memorex ATL (IS) 5-63
- MINDEX (IS) 5-21
- MINRET (IS) 5-21
- MINRET (UR) 1-4
- MODE (IS) 5-21
- MODE (UR) 1-30
- MSG088 (IS) 5-22
- MULTCPU (IS) 2-68, 5-22
- NCATDSK (IS) 5-22
- NCATDSK (UR) 1-4, 1-20
- NOGVIS (IS) 5-23
- ONLREWCD (IS) 5-23
- OPTI-TAPE ACL support (IS) 5-73
- overriding (IS) 6-27
- PASSWD (IS) 5-23
- PREFIX (IS) 5-23
- PURBYP (IS) 5-23
- PURBYP (UR) 1-26, 6-17
- RECORD (IS) 5-23
- RECORD (UR) 1-27, 11-1
- Recorder File (UR) 1-27
- RECSHR (IS) 5-23
- RECSHR (UR) 1-27
- RECWARN (UR) 1-27
- RECWMSG (UR) 1-27
- RERUN (IS) 5-23
- RETEN (IS) 5-24
- RETEN (UR) 1-20
- RMNTPRT (IS) 5-24
- RMNTPRT (UR) 12-16
- SHARE (IS) 5-24
- SHARE (UR) 1-29
- SLASIZE (IS) 5-24
- special (IS) 5-31
- Start Track 1 datasets (UR) 1-4
- STK ACL support (IS) 5-68
- STRTRK (IS) 5-24
- STRTRK (UR) 1-4, 13-5
- syntax (IS) 5-4
- TAPCUU (IS) 5-24
- TAPE (IS) 5-24
- TAPLUB (IS) 5-24
- TAPPOL (IS) 2-44, 5-25
- TPAMODE (IS) 5-25
- TSTJOB (IS) 5-26
- UNCDSK (IS) 5-26
- UNCDSK (UR) 1-5, 6-30
- UNCTAP (IS) 5-26
- UNCTAP (UR) 1-5, 1-29
- UNCTL (IS) 5-26
- UNCTL (UR) 1-5
- UNLABEL (IS) 5-26
- UNLOAD (IS) 5-26
- VGSFTAPE (IS) 5-27
- VM (IS) 5-27
- VM (UR) 9-20
- VSAM (IS) 5-28
- VSAM (UR) 1-5, 7-28
- worksheet (IS) 1-12
- Configuration options (IS) **5-8**
- Configuration options (UR) 1-2
- Confirmation options (UR) 2-15
- Console (IS) 6-29
- CONTIG parameter (UR) 6-30
- Continuing with a dependent task (UR) 10-6
- Controlled datasets (UR) 1-3
- Controlled tapes (UR) 2-40, 2-41, 8-14
- Conversion issues (IS) 2-8
- Converting
 - to BIM-EPIC See BIM-EPIC Conversion Guide
 - uncontrolled to controlled (UR) 13-5
- Copying and merging datasets (UR) 8-9
- CP command table (IS) 3-31
- CP commands (UR) 9-20
- CP nucleus
 - Autoattach installation (IS) 3-1
 - backing up (IS) 3-28
- CP nucleus (IS) 3-18, 3-26
- CPU independence
 - disk pools (UR) 1-14, 1-31
- CPU independence (UR) **7-30**
- CPU/partition-independent pools (UR) 1-31
- CPUID (IS) 6-15
- CPUIDS configuration option (IS) 5-15
- Create Date/Time Activity Report (UR) A-24
- Create Job/Time Activity Report (UR) A-13
- Creating
 - EDD (UR) 3-12
 - EDDs (UR) 2-23
 - ERD (IS) 2-34
 - master index (IS) 2-23
 - Recorder History dataset (UR) 11-2
 - security profiles (UR) 2-8

- tape volume labels (UR) 1-30
- Creation date
 - changing (UR) 3-37
 - report listing (UR) A-30
- Creation date (UR) 1-18
- Creation job
 - changing (UR) 3-37
 - report listing (UR) A-30
- Creation job (UR) 1-18
- Creation time (UR) 1-18
- Cross-Reference Report (UR) 12-23
- Cross-Reference Reports
 - input to TSIDBRP (UR) 11-3
- Cross-Reference Reports (UR) A-8
- CTL control statement (IS) 2-32
- Current condition code See Job management control
 - language
- Current utilization, displaying (UR) 3-11
- Customer support xii, xi
- Customized data displays (UR) 2-12
- Customizing
 - BIM-EPIC processing (UR) 1-2
 - confirmation options (UR) 2-15
 - EDD display format (UR) 2-13
 - version display format (UR) 2-14
- CWDS See Common work datasets
- CWDSPOL configuration option (IS) 5-15
- CYCL configuration option (IS) 5-15
- CYCL configuration option (UR) 1-20
- Cycle a Version panel (UR) 2-50
- CYCLE command
 - difference from RETAIN (UR) 2-50
 - versions (UR) 2-50
- CYCLE command (UR) 2-5, 2-51, 3-20
- Cycles retention
 - overriding (UR) 3-20
- Cycles retention (UR) 1-11
- CYL parameter (UR) 6-30

D

- D option (UR) 5-10
- DA datasets
 - secondary allocation unavailable (UR) 13-3
- DA datasets (IS) 5-37
- DA datasets (UR) 1-5
- Daily maintenance JCL (IS) 2-91
- DAILYn SAMPJOB (IS) 4-2
- DATA parameter (UR) 6-39, 9-8
- Database utilities (UR) 7-35
- DATA-REPORTER (IS) 5-31
- Dataset defaults

- overriding (UR) 6-9
- Dataset definitions See BIM-EPIC dataset definitions
- Dataset drop
 - TLBL option 16 (UR) 5-7
- Dataset drop (UR) 6-3, 7-63
- Dataset fragmentation (UR) 13-2
- Dataset name
 - DSN DD parameter (UR) 6-22
 - specifying on DD statement (UR) 6-7
- Dataset name (UR) 1-9
- Dataset names
 - changing (UR) 3-36
- Dataset to Tape Report (UR) 12-24, A-11
- DATASET.NAME (UR) 1-43
- dataset-name control statement (UR) 13-29
- Datasets
 - assigning vaulting methods (UR) 12-19
 - cataloging (UR) 1-3
 - changing retention (UR) 1-25
 - concatenating (UR) 6-4
 - concatenating with DLBL option E (UR) 5-10
 - concatenating with multiple opens (UR) 5-12
 - control (UR) 1-5
 - copying and merging (UR) 8-9
 - defining (UR) 7-5
 - deleting (UR) 5-10
 - direct access (UR) 7-25
 - expiration rules (UR) 1-22
 - generation or work (UR) 1-10
 - ISAM (UR) 7-26
 - naming (UR) 1-9
 - onloading (UR) 13-27
 - renaming (UR) 3-36
 - restoring (UR) 13-27
 - revolving (UR) 12-3
 - rules for naming (UR) 1-9
 - scratching (UR) 1-35
 - sequential (UR) 7-25
 - Start Track 1 (UR) 1-4
 - uncontrolled (UR) 2-41
 - uncontrolled disk (UR) 6-30
- Date formats
 - European (IS) 5-19
- Date time activity report
 - creating (UR) A-24
- Days after last access (DLA) (UR) 1-11
- DCB parameter (UR) 6-7
- DCT entries (IS) 2-88
- DD names
 - special (UR) 6-5
- DD names (UR) 6-2
- DD statements

- ABSTR parameter (UR) 6-30
- additional parameters (UR) 6-2
- allocation by logical record (UR) 6-30
- ALX parameter (UR) 6-30
- AMORG parameter (UR) 6-12
- AMP (UR) 6-7
- AMP parameter (UR) 6-12
- BLKSIZE parameter (UR) 6-13
- BUFSP parameter (UR) 6-12
- catalog override example (UR) 6-9
- cataloging (UR) 7-18
- CATLG parameter (UR) 6-17
- CLOSE parameter (UR) 6-24
- CONDCAT parameter (UR) 6-17, 7-64
- CONTIG parameter (UR) 6-30
- continuations (UR) 6-3
- CYL parameter (UR) 6-30
- DATA parameter (UR) 6-39
- dataset drop (UR) 7-63
- dataset fragmentation (UR) 13-2
- DCB (UR) 6-7
- DCB parameter (UR) 6-13
- DD name (UR) 6-2
- default version (UR) 7-4
- DELETE parameter (UR) 6-17
- DEN parameter (UR) 6-13
- DISP (UR) 6-7
- DISP parameter (UR) 6-15
- DROP option (UR) 6-3
- DSN parameter (UR) 6-7, 6-22
- DSORG parameter (UR) 6-14
- dummy datasets (UR) 6-41
- END parameter (UR) 6-24
- fragmentation (UR) 6-30
- FREE parameter (UR) 1-30, 6-8, 6-24
- ignoring datasets (UR) 6-41
- JOB CAT (UR) 6-5
- KEEP parameter (UR) 6-17
- LABEL (UR) 6-8
- LABEL parameter (UR) 6-25
- LRECL parameter (UR) 6-14
- MOD subparameter (UR) 6-15
- multi-dataset coding (UR) 6-6
- MXIG parameter (UR) 6-30
- NEW subparameter (UR) 6-15
- OLD subparameter (UR) 6-15
- OPTION (UR) 6-8, 6-26
- OPTION parameter (UR) 6-38
- OS concatenation (UR) 6-3
- parameters (UR) 6-7
- PASS parameter (UR) 6-17
- pool request (UR) 6-35
- printer and punch datasets (UR) 6-38
- PRIVATE parameter (UR) 6-35
- REF (UR) 6-35
- RETAIN parameter (UR) 6-35
- RLSE parameter (UR) 6-30
- ROUND parameter (UR) 6-30
- secondary allocation (UR) 6-30
- SER parameter (UR) 6-35
- SHR subparameter (UR) 6-15
- SPACE (UR) 6-8, 6-30
- SPACE parameter (UR) 6-27, 13-4
- special ddnames (UR) 6-5
- specifying (UR) 6-2, 6-9
- specifying for tape (UR) 6-9
- Start Track 1 example (UR) 6-9
- STEP CAT (UR) 6-5
- syntax checking (UR) 6-3
- SYSABEND (UR) 6-5, 6-6
- SYSIN (UR) 6-5
- SYSMDUMP (UR) 6-5, 6-6
- SYSnnn parameter (UR) 6-27
- SYSOUT parameter (UR) 6-38
- SYSUDUMP (UR) 6-5, 6-6
- TRK parameter (UR) 6-30
- uncontrolled dataset example (UR) 6-9
- uncontrolled datasets (UR) 6-9
- uncontrolled disk (UR) 6-30
- UNIT (UR) 6-8
- UNIT parameter (UR) 6-7, 6-33, 6-38
- unsupported ddnames (UR) 6-5
- VOL parameter (UR) 6-7
- volser request (UR) 6-35
- VOLUME (UR) 6-8
- VOLUME parameter (UR) 6-35
- VSAM example (UR) 6-10
- VSAM processing (UR) 6-10, 6-26
- VSAM/SAM processing (UR) 6-10
- DD statements (UR) **6-1**
- DDERR configuration option (IS) 5-16
- DDGEN configuration option (IS) 5-16
- DDGEN configuration option (UR) 6-17, 6-22
- Deactivating
 - BIM-EPIC (IS) 5-44
 - IBM ACL support (IS) 5-60
 - Memorex ATL support (IS) 5-63
 - QDAS (IS) 5-52
 - Recorder (IS) 5-45
 - STK ACL support (IS) 5-68
 - TPRT transaction (UR) 12-16
 - TSIDBPR (UR) 12-16
- DEBUG configuration option (IS) 5-17
- Debugging (IS) 6-24

- Default profiles
 - \$DEFPROF (UR) 2-12
 - \$DEFSEC (UR) 2-7
 - \$EPIC (UR) 2-7, 2-12
 - user (UR) 2-12
- Defaults
 - density (UR) 1-12
 - disk pool (UR) 1-19
 - pools (UR) 1-14
 - security profiles (UR) 2-7
- DEFEXT configuration option (IS) 5-17
- DEFEXT configuration option (UR) 1-20
- Define an EDD panel (UR) 2-23
- DEFINE command (UR) 2-5
- Defining
 - datasets (UR) 7-5
 - EDDs (UR) 2-23
 - vault locations (IS) 6-39
 - vaulting methods (IS) 6-40
- DEFPOL configuration option (IS) 5-17
- DEFPOL configuration option (UR) 1-14, 1-20
- DEFTPL configuration option (IS) 5-17
- DEFWPOL configuration option (IS) 5-17
- DEL control statement (IS) 2-33
- DELETE
 - DD parameter (UR) 6-17
- DELETE command
 - TSIDVUT (IS) 6-58
- DELETE command (UR) 2-5, 2-29, 2-39, 3-22
- Deleting
 - EDD (UR) 2-29, 3-22, 3-33
 - EDDs (UR) 1-35
 - versions (UR) 3-33
 - volsers (UR) 3-33
 - VTOC entries (IS) 6-58
- Deleting datasets
 - DD statements (UR) 6-17
 - DELETE DD parameter (UR) 6-17
 - DLBL option D (UR) 5-10
 - TSIDDEL (UR) 8-7
- Density
 - DEN DD parameter (UR) 6-13
- DEQ command (UR) 3-23
- DETACH (UR) 12-4
- Detail display (UR) A-17
- Detail records
 - ADDRESS EPIC (UR) 4-12
- Detail records (UR) 4-5
- Detail report
 - TSIDMNT (UR) A-17
- Device address (UR) 1-18
- Device down (UR) 3-31
- Device type request
 - DD statements (UR) 6-33
- Device type, choosing (UR) 7-56
- DFSORT (IS) 2-10
- DFSORT (UR) 7-37
- DHLDOPEN (IS) 6-15
- DHLDOPEN (UR) 1-22, 1-38
- Direct access datasets
 - DD statements (UR) 6-14
- Direct access datasets (UR) 7-25
- Disk allocation
 - aligning on cylinder boundary (UR) 5-10
 - defining volumes to QDAS (IS) 2-55
 - leveling (IS) 2-43
- Disk datasets
 - close processing (UR) 1-43
 - converting to controlled (UR) 13-5
 - grouping (UR) 1-9
 - names (UR) 1-9
 - open processing (UR) 1-42
 - placing on specific volumes (UR) 13-6
 - removing from BIM-EPIC control (IS) 6-66
 - scratching (UR) 1-35, 1-38
- Disk pool name (UR) 1-14
- Disk pools
 - CPU-independent (UR) 1-14, 1-31
 - creating (IS) 2-39, 2-42
 - default pool (UR) 1-19
 - defining (IS) 2-43
 - design (IS) 2-40
 - leveling (IS) 2-43
 - leveling (UR) 1-31
 - locating (UR) 1-42
 - partition-independent (UR) 1-14, 1-31
- Disk pools (UR) 1-28, **1-31**
- Disk space
 - freeing (UR) 3-22
- Disk to tape (UR) 7-56
- Disk volumes
 - dismounting (UR) 3-31
 - sharing (IS) 2-52, 2-68
- DISKALL configuration option (IS) 5-17
- Dismounting a disk volume (UR) 3-31
- DISP parameter (UR) 6-7, 6-15
- DISP=(x,CATLG) (UR) 1-19
- Dispatcher
 - EP commands (IS) 5-53
- Dispatcher (IS) 5-53
- Display active report listing (UR) A-30
- DISPLAY command (UR) 2-21, 3-24
- Display NEW report listing (UR) A-30
- Display PULL report listing (UR) A-30

- Display PULL, DSN report listing (UR) A-30
- Display scratch report listing (UR) A-33
- Display vault report listing (UR) A-30
- Displaying
 - DSN Catalog entries (UR) 3-24, 4-2
 - EDDs (UR) 2-19, 2-22
 - online VTOC (UR) 13-11
 - tape information (UR) 2-42
 - tapes required by a job (IS) 6-53
 - version attributes (UR) 2-33
 - version information (UR) 2-30
 - VTOC entries (IS) 6-56
 - VTOC entries (UR) 1-31, A-15
- Distribution tape
 - loading library modules (IS) 2-12
- DITTO (IS) 5-32
- DITTO (UR) 7-35, 7-47
- DL/1 datasets (UR) 13-4
- DLAHOLD (IS) 6-15
- DLAUPD= (UR) 13-23
- DLBL option codes
 - A align on cylinder boundary (UR) 5-10
 - B=n reblocking (UR) 5-10, 8-9
 - C conditional catalog (UR) 5-10, 7-64
 - D delete after closed as input (UR) 5-10
 - E concatenation with one open (UR) 5-10
 - F fragmentation (UR) 5-10
 - G=n access absolute generation (UR) 5-10
 - H override FRAG=YES (UR) 5-10
 - I override BIM-EPIC control (UR) 5-11
 - L no LUB allocation (UR) 5-11
 - M issue allocation messages (UR) 5-11, 6-27
 - N inhibit truncation feature (UR) 5-11
 - O use EXTENT size (UR) 5-11
 - P purge at EOJ (UR) 5-11
 - R=nnn allocation by logical records (UR) 5-11
 - S reuse same space (UR) 5-12
 - T dataset drop (UR) 5-12, 7-63, 8-9
 - V=n access a relative version (UR) 5-12
 - X delete at close or EOJ (UR) 5-12
- DLBL option codes (UR) **5-10**
- DLBL statements
 - automatic truncation (UR) 13-4
 - BIM-EPIC enhancements (UR) 5-8
 - default version (UR) 7-4
 - F option (UR) 13-2
 - inhibiting truncation (UR) 13-4
- DLBL statements (UR) **5-8**
- Do not rewind (UR) 5-7
- Doctor D (IS) 5-31
- Documentation xiv, xiii
- DOS cancel code See Job management control language
- DOS label See VSE label
- DOSCC parameter (UR) 9-9
- Drive assignments (UR) 1-29
- DROP options
 - DD statements (UR) 6-3
- DSKLUB configuration option
 - assigning SYS number (UR) 1-28
- DSKLUB configuration option (IS) 5-18
- DSKRET configuration option (IS) 5-18
- DSKRET configuration option (UR) 1-24
- DSN Catalog
 - accessing (UR) 4-5, 4-11
 - capacity (UR) 3-11
 - Combining (IS) 6-70
 - controlled tapes (UR) 2-40
 - creating (IS) 2-17
 - daily maintenance JCL (IS) 2-91
 - dataset name sequence (UR) A-11
 - dequeuing (UR) 3-23
 - EDD information (UR) 1-6
 - efficiency (IS) 2-23
 - efficiency (UR) 1-4
 - emergency recovery (IS) 4-8
 - enlarging (IS) 6-13
 - enqueueing (UR) 3-27
 - entries (UR) 1-6
 - fast backup (IS) 6-10
 - fast restore (IS) 6-11
 - formatting (IS) 2-23, 6-21
 - IJSYSDS (IS) 5-2
 - in volume sequence (UR) A-23
 - labels (IS) 2-58
 - location (IS) 2-19
 - maintaining (IS) 6-2
 - maintenance (UR) A-5
 - managing in batch mode (UR) 3-1
 - master index (IS) 2-23
 - moving or changing size (IS) 4-6
 - online management See also Online Manager
 - rebuilding (IS) 6-23
 - record structure (UR) 4-5
 - removing versions (UR) 1-35
 - resequencing (UR) 4-5
 - scratching datasets automatically (IS) 6-14
 - searching (UR) 2-19
 - sequential and random access (UR) 4-5
 - sharing (IS) 2-68
 - size requirements (IS) 2-19
 - statistics (UR) 2-5
 - upgrading from version 2.x (IS) 2-92

- version information (UR) 1-6
- DSN Catalog (UR) 1-3
- DSN Catalog entries (UR) 1-3, 2-41
- DSN DD parameter (UR) 6-22
- DSN parameter (UR) 6-7
- DSN/JOB Cross-Reference Report (UR) 11-6, A-8
- DSNAME (IS) 6-11
- DSNAME configuration option (IS) 5-18
- DSNWARN configuration option (IS) 5-18
- DSNWMSG configuration option (IS) 5-18
- DSORG DD parameter (UR) 6-14
- DSPLY command
 - VAULT (UR) 12-21
 - VLT (UR) 12-21
- DSPLY command (IS) 6-7
- DSPLY command (UR) 4-2
- DTF names
 - default dataset name (IS) 5-32
 - excluding from BIM-EPIC control (UR) 1-5
 - in TLBLs (UR) 5-2
 - inhibiting LUB allocation (IS) 2-47, 2-48
 - RFBKUP (IS) 5-49
 - TSIDDMP (UR) 8-16
 - TSIDDTD (UR) 8-9
 - uncontrolled (IS) 1-11, 2-47
 - unit record (UR) 6-2
- DTFCD (UR) 6-2, 7-56
- DTFCP (UR) 7-56
- DTFDI (UR) 7-56
- DTFPR (UR) 6-2, 7-56
- DTFs
 - TYPEFLE=WORK (UR) 7-32
- DUMMY parameter (UR) 6-41
- Dummy tape device (IS) 2-61
- DUSP (IS) 2-11, 5-31
- DVCDN (UR) 3-31
- Dynamic space GETVIS (IS) 2-63

E

- E option (UR) 5-10
- Early Drive Release (UR) 1-30
- EDD *See* BIM-EPIC dataset definitions
- EDD Information panel (UR) 2-22
- Efficiency considerations
 - disk allocation (UR) 1-31
 - master index (IS) 2-23
 - Start Track 1 datasets (UR) 1-4
- Emergency catalog recovery (IS) 4-8
- Emergency recovery
 - rebuilding DSN Catalog (IS) 6-23
- Emergency support xiii, xii

- Encryption password (UR) 1-9
- END control statement (IS) 5-48
- END parameter (IS) 2-48
- END parameter (UR) 6-24
- End-of-extent processing
 - disk (UR) 1-42
- End-of-volume processing (UR) 1-40
- ENQ command (UR) 3-27
- EOF extent (IS) 5-31
- EOF extent (UR) 1-17
- EP commands
 - Dispatcher (IS) 5-53
- EP009 message
 - ACCEPT response (UR) 11-6
- EP038 message
 - DISPLAY response (IS) 6-60
 - FRAG response (UR) 5-10, 13-2
- EP930 message (UR) 12-5
- EPAT conversions *See* BIM-EPIC Conversion Guide
- EPIC transaction (IS) 2-83
- EPICAA PARTCAT file (IS) 3-23
- EPICFLOW command (UR) 2-16
- EPICOFF (IS) 6-26
- EPICOFF configuration option (IS) 5-18
- EPICON (IS) 5-52, 6-26
- EPICON procedure (IS) 5-2
- EPICREC.OAL (UR) 4-12
- EPICTAPEPOL-x (IS) 5-72
- EPLI macro (IS) 6-51
- EPS022 special option (UR) 7-47
- EPS043 special option (UR) 12-4
- EPVAULT *See* Vaulting
- ERD *See* BIM-EPIC Resource Dataset
- ERDFILE configuration option (IS) 2-25, 5-19
- EURODAT configuration option (IS) 5-19
- European date format (IS) 5-19
- Event posting
 - executing in batch (UR) 10-8
 - releasing TSIDREL AND TSIDCREL (UR) 10-4
 - SYSPARM value (UR) 10-8
 - TSIDCHK and TSIDCCHK (UR) 10-6
 - TSIDLOC and TSIDCLOC (UR) 10-5
 - TSIDUNL and TSIDCUNL (UR) 10-7
- Event posting (UR) 10-11
- Event posting entries (IS) 2-78
- Examples
 - SPACE parameter (UR) 6-30
- EXCLUDE (UR) 11-3
- Excluding
 - DTF names (UR) 1-5
- Executing
 - in batch (UR) 10-8

- in CICS (UR) 10-11
- standard processing subroutines (UR) 10-9
- Exiting
 - Online Manager (UR) 2-4
 - TOJA (UR) 11-15
- Expiration
 - LABEL parameter (UR) 6-25
- Expiration date
 - changing (UR) 3-37
 - overriding (UR) 1-25
 - report listing (UR) A-30
- EXPORT (UR) 7-45
- EXTEND/VSE
 - start before BIM-EPIC (IS) 2-8
- Extending a tape
 - MOD subparameter (UR) 6-15
- Extent statement
 - reusing space (UR) 7-10
- Extent statements
 - overriding (UR) 7-8
- EXTENT statements
 - controlled dataset (UR) 7-25
 - overriding EDD size (UR) 5-11
 - required with DLBL option I (UR) 5-11
 - Start Track 1 dataset (UR) 7-25
 - Start Track 1 datasets (UR) 1-4

F

- F option (UR) 5-10
- FASTBU command (IS) 6-10
- FASTRS
 - restrictions (IS) 6-11
- FASTRS command (IS) 6-11
- FAVER (UR) 7-43
- FBA CFSIZE (UR) 6-26
- FBAFACT configuration option (IS) 5-19
- FCOBOL (IS) 2-5
- FCT Table (IS) 2-87
- Field help (UR) 2-16
- First-time installation (IS) 2-4
- FMT control statement (IS) 2-34
- Forced initialization (UR) 12-10
- Formatting
 - ERD (IS) 2-34
- FORTTRAN (UR) 7-35
- FRAG configuration option
 - overriding with DLBL option H (UR) 5-10
- FRAG configuration option (IS) 5-19
- FRAG configuration option (UR) 13-2
- Fragmentation
 - DD statements (UR) 6-30

- DLBL option F (UR) 5-10
- FREE command (UR) 12-6
- FREE DD parameter (UR) 6-8
- FREE parameter (UR) 1-30, 6-24
- Free records, number of (UR) 3-11

G

- G option (UR) 5-10
- GDI
 - example (UR) 7-59
- GDI (UR) 7-56
- GDI configuration option
 - TSIDDMP (UR) 8-16
 - TSIDDTD (UR) 8-9
- GDI configuration option (IS) 5-19
- GDI configuration option (UR) 8-6
- GDI/GDIUR (UR) 7-56
- GDIUR
 - DD statement reset (UR) 6-26
 - ending (UR) 6-27
- GDIUR (UR) 7-56
- GDUIR configuration option (IS) 5-20
- Generation datasets
 - autocatalog with DD statements (UR) 6-17, 6-22
 - dataset name in JCL (UR) 7-6
 - DD statements (UR) 6-22
- Generation datasets (UR) **1-10**
- Generation number (UR) 1-16
- GETVIS
 - dynamic space (IS) 2-63
 - partition size requirements (IS) 2-63
 - releasing (UR) 4-9
 - system (IS) 2-63
- Global constants See Job management control language
- GOTO parameter (UR) 9-7, 9-9
- GSMETA (IS) 2-9

H

- H option (UR) 5-10
- HCPMDLAT macro (IS) 3-29, 3-37
- HDR1 labels
 - expiration (UR) 12-2
 - retention (UR) 12-2
- Help, online (UR) 2-16
- Highest condition code See Job management control language
- High-level assembler (IS) 3-31
- HOLD parameter (UR) 9-5
- Holding assignment
 - FREE statement (UR) 6-24

Holding drive assignment with TLBL option 8 (UR) 5-7

HSC pool definitions (IS) 5-72

I

I option (UR) 5-11

IBL ACL

requirements (IS) 2-5

IBM ACL

ERD (IS) 2-25

IBM ACL support

activating and deactivating (IS) 5-60

IBM utilities

checkpoint/restart (UR) 7-54

DITTO (UR) 7-47

IBM utilities (UR) 7-35

IDCAMS

backup and restore (UR) 7-43

example (UR) 6-27

import and export (UR) 7-45

REPRO (UR) 7-44

IDCAMS (UR) 7-35

IGNORE (UR) 7-60

Ignoring datasets

DD statements (UR) 6-41

IJSYS files (UR) 7-7

IJSYSDR (IS) 2-58, 5-45, 6-21

IJSYSDS (IS) 2-58, 5-2, 6-10, 6-11, 6-21

IJSYSJQ (IS) 2-58

IJSYSNR (UR) 1-5

IJSYSPC (UR) 1-5

IJSYSSP (IS) 2-58

IJSYSxx datasets (UR) 7-33

ILIST parameter

excluding DTF names (UR) 1-5

ILIST parameter (IS) 2-48

IMODs (UR) 4-12

IMODTBP (UR) 7-43

IMPORT (UR) 7-45

INCLUDE

TSIDOFD (UR) 13-20

Increasing

partition GETVIS (IS) 2-63

system GETVIS (IS) 2-63

INDEX (IS) 6-12

INDEX (UR) 4-5

Index, resequencing (UR) 4-5

Indexed sequential

DD statements (UR) 6-14, 6-22, 6-30

Inhibiting LUB allocation

DLBL option L (UR) 5-11

TLBL option 64 (UR) 5-7

with SYS number (UR) 1-10

Inhibiting LUB allocation (UR) 7-7

Inhibiting truncation (UR) 5-11

Initial Installation Checklist (IS) 1-2

Initializing tapes (UR) 1-30, 2-40, 12-9

Initiating

subtasks (IS) 5-53

Input DTF names

TSIDDM (UR) 8-16

TSIDDTD (UR) 8-9

Installation

BIM-GSS (IS) 2-2

first-time (IS) 2-4

for conversions *See* BIM-EPIC Conversion Guide

Installation files

VM/ESA Version 1 Release 1.1 (IS) 3-27

VM/ESA Version 1 Release 2 (IS) 3-20

VM/ESA Version 2 Release 2 (IS) 3-10

Installation procedures (IS) 2-15

Installation requirements

VM/ESA 1.1 (IS) 3-27

Installing CICS (IS) 2-78

Interactive Partition GETVIS VSE/ICCF (UR) 7-52

IOMDTRT (UR) 7-43

ISAM *See* Indexed sequential

ISAM datasets (UR) 1-5, 7-26

IUCV connection (IS) 5-65, 5-70

IUCV Path (IS) 5-65, 5-70

J

JCL

BIM-EPIC enhancements (UR) 1-39

BIM-EPIC maintenance (IS) 2-91

DD statements (UR) 6-9

specifying DD statements (UR) 6-2

JCL enhancements (UR) 7-15

JCLAC configuration option (IS) 5-20

JCLLANG configuration option (IS) 5-20

JCLLANG configuration option (UR) 9-1

JCLOPT configuration option

TLBL alpha options (UR) 5-3

JCLOPT configuration option (IS) 5-20

JCLOVRDE configuration option (IS) 5-20

JCLOVRDE configuration option (UR) 6-9, 6-17

JMCL *See* Job management control language

SCAN (UR) 6-3

JMCL parameters

AND (UR) 9-9

DATA (UR) 9-8

DOSCC (UR) 9-9

- GOTO (UR) 9-7, 9-9
- HOLD (UR) 9-5
- NAME (UR) 9-7
- OR (UR) 9-9
- RESET (UR) 9-13
- SETC (UR) 9-11
- SETG (UR) 9-11
- TIME (UR) 9-5
- TYPRUN (UR) 9-6
- Job accounting
 - activation (IS) 2-94
 - by CPU (UR) A-24
 - by CPU and by job (UR) A-13
 - by date and time (UR) A-24
 - database (UR) 11-13
 - database creation (IS) 2-85
 - installation (IS) 2-85
 - online installation (IS) 2-84
 - POWER (UR) 7-48
 - supervisor requirements (IS) 2-5
 - TOJA installation (IS) 2-87
 - TSIDJAC (UR) 11-8
 - TSIDJLD (UR) 11-10
 - TSIDTAC (UR) 11-9
- Job accounting database (UR) A-3
- Job cancellation (UR) 1-38
- Job execution
 - TIME limit (UR) 9-5
- Job management control language
 - &n (UR) 9-12
 - condition checking and branching (UR) 9-7
 - conditions and IF (UR) 9-9
 - CP commands (UR) 9-20
 - CPUID (UR) 9-9
 - current time (UR) 9-9
 - current time date (UR) 9-9
 - DOS cancel code (UR) 9-9
 - global constants (UR) 9-9
 - highest condition code (UR) 9-9
 - job cancel options and syntax checking (UR) 9-5
 - parameters (UR) 9-2
 - partition ID (UR) 9-9
 - preventing EOJ after cancel (UR) 9-5
 - symbolic replacements (UR) 9-12
- Job management control language (UR) **9-1**
- Job Overview panel (UR) 11-13, A-3
- Job scheduling
 - programs (IS) 6-50
 - TSIDFAV macro (IS) 6-44
- Job scheduling (UR) 10-4
- Job Selection panel (UR) 11-13
- Job Time Activity Report (UR) A-13

- JOBACCT configuration option
 - required with DLBL option P (UR) 5-11
 - required with DLBL option X (UR) 5-12
- JOBACCT configuration option (IS) 5-20
- JOBACCT configuration option (UR) 11-1
- JOBCAT statements
 - overriding (UR) 6-27
- JOBCAT statements (UR) 6-5
- JOBLIB statements
 - unsupported (UR) 6-5
- JOBQ
 - creating (IS) 2-17
 - location (IS) 2-24
- Jobs
 - running (UR) 7-1
- Jobs in the POWER queues (UR) 10-4
- JQNAME configuration option (IS) 5-20
- JQWARN configuration option (IS) 5-21
- JQWMSG configuration option (IS) 5-21

K

- KEEP DD parameter (UR) 6-17

L

- L option (UR) 5-11
- Label creation (UR) 12-16
- LABEL DD parameter (UR) 6-8
- LABEL parameter
 - retention days (UR) 6-25
- LABEL parameter (UR) 6-25
- Label printing
 - installation (IS) 2-84
 - sample Assembler batch subroutine (IS) 6-46
 - sample Assembler online subroutine (IS) 6-47
 - sample COBOL batch subroutine (IS) 6-46
 - sample COBOL online subroutine (IS) 6-47
- Labels
 - batch printing (IS) 6-46
 - online printing (IS) 6-47
 - print subroutine (IS) 6-45
- Labels, disk (UR) 12-2
- Labels, tape See Tape labels
- Large tapes (UR) 12-3
- Leveling a disk pool (UR) 1-31
- LIBDEF procedures (IS) 2-64
- LIBR (UR) 7-46
- Library
 - modules (IS) 2-12
 - search chains (IS) 2-64
- License code
 - in activation procedure (IS) 2-94

Listing tapes to be moved (UR) A-35
 Listing the datasets to be backed up (UR) A-19
 Listing the datasets to be offloaded (UR) A-19
 LMS pool definitions (IS) 5-67
 Local Version Vector Table (IS) 3-15, 3-23
 LOCALMOD disk (IS) 3-10, 3-20, 3-27
 LOCK command (UR) 2-5, 2-26, 2-35, 3-28
 Locked status (UR) 1-16
 Locking
 EDD (UR) 2-26
 version (UR) 2-35
 Locking a dataset or version (UR) 3-28
 LOG option (UR) 3-4
 Logging
 VSAM datasets (UR) 7-28
 Logical IOCS (UR) 1-17
 LOGSRC option (UR) 5-4
 LUB allocation
 direct access files (IS) 5-37
 DTFPH (UR) 7-7
 inhibiting (IS) 2-47, 2-48
 inhibiting with DD statements (UR) 6-27
 inhibiting with DLBL option L (UR) 5-11
 inhibiting with SYS number (UR) 1-10
 inhibiting with TLBL option 64 (UR) 5-7
 physical IOCS (UR) 7-7
 SORTIN/SORTOUT (IS) 5-31
 SORTWK (IS) 5-31
 TSIDASS (UR) 7-7
 TSIDMAP (UR) 7-7
 UIN/UOUT (IS) 5-31
 warning (UR) 7-7
 LUB allocation (UR) 1-28, 7-7

M

M option (UR) 5-11
 Main vault (UR) 3-30
 MAINT
 multi-CPU environment (IS) 5-46
 MAINT control statement (IS) 5-49
 Maintenance JCL (IS) 2-91
 Maintenance procedures
 DSN Catalog (IS) 6-2
 Maintenance procedures (IS) 4-2, 4-7
 Manage EDDs panel (UR) 2-21
 Manage Tape Volumes panel (UR) 2-43
 Manage Versions panel (UR) 2-32
 Managing
 EDDs (UR) 2-21
 tape information (UR) 2-43
 vaults (UR) 12-17

 versions (UR) 2-30
 Manual cataloging
 explanation (UR) 1-7
 Manual cataloging (UR) 2-2
 Manually controlled vaults (UR) 12-22
 Master index
 MINDEX configuration option (IS) 5-21
 Master index (IS) 2-23
 Master records
 ADDRESS EPIC (UR) 4-11
 Master records (UR) 4-5
 MAX block size (UR) 6-13
 MAXBACK (IS) 2-11, 5-31
 Maximum secondaries (UR) 1-14
 MCAT alias (UR) 3-29
 MCAT status (UR) 1-16
 MCATALOG command (UR) 2-5, 3-29
 Memorex ACL
 ERD (IS) 2-25
 requirements (IS) 2-5
 Memorex ATL
 configuration options (IS) 5-63
 LMS definitions (IS) 5-67
 scratch pools (IS) 6-7
 Memorex ATL support
 activating and deactivating (IS) 5-63
 Merging datasets, TSIDDTD (UR) 8-9
 Messages Also see Messages and Codes
 allocation (UR) 6-26
 TSIDVUT (IS) 6-56
 METHOD macro Also see Vaulting
 assembling and linking (IS) 6-41
 METHOD macro (IS) **6-40**
 METHOD macro (UR) 12-17
 Methods, vaulting (IS) 6-40
 MINDEX configuration option (IS) 5-21
 MINRET configuration option (IS) 5-21
 MINRET configuration option (UR) 1-4
 MOD subparameter (UR) 6-15
 MODE configuration option
 default density (UR) 1-12
 MODE configuration option (IS) 5-21
 MODE configuration option (UR) 1-30
 Mode setting
 DEN DD parameter (UR) 6-13
 Modeling
 security profiles (UR) 2-8
 user profiles (UR) 2-12
 Modifying
 confirmation options (UR) 2-15
 EDD display format (UR) 2-13
 EDD information (UR) 2-25

- expiration date (UR) 3-37
- security profiles (UR) 2-8
- standard labels (IS) 2-58
- tape modes (IS) 2-61
- version display format (UR) 2-14
- Modifying VSE labels
 - TSIDGLB and TSIDPLB (UR) 8-21
- Mounting tapes for output
 - uncontrolled (UR) 12-7
- MOVE command (UR) 2-43, 3-30
- MOVESER control statement (UR) 12-20
- MOVESLT control statement (UR) 12-20
- Moving
 - JCL (UR) 6-1
- Moving the DSN Catalog See Maintenance procedures
- Moving the Recorder File See Maintenance procedures
- MSES/E service procedures (IS) 3-27
- MSG088 configuration option (IS) 5-22
- MTC (UR) 7-12
- MULTCPU configuration option
 - CPU independence (UR) 6-23
- MULTCPU configuration option (IS) 2-68, 5-22
- MULTCPU configuration option (UR) 3-27
- Multi-CPU environment (IS) 5-46, 5-49, 5-63, 5-68
- Multi-dataset groups
 - prime dataset name (UR) 1-12
- Multi-dataset processing
 - autocatalog with DD statements (UR) 6-17
 - examples (UR) 7-18
 - FREE DD statement (UR) 6-24
 - refer-back with DD statements (UR) 6-35
 - scratching multi-dataset tapes (UR) 7-15
- Multiple job exits (IS) 2-65
- Multi-volume
 - DD statements (UR) 6-35
 - processing limits (UR) 7-13
- MVS
 - compatibility (UR) 6-3, 6-33
 - dataset drop not valid under (UR) 6-3
 - JCL (UR) 6-1
- MXIG parameter (UR) 6-30

N

- N option (IS) 5-31
- N option (UR) 5-11
- NAME parameter (UR) 9-7
- NCATDSK configuration option (IS) 5-22
- NCATDSK configuration option (UR) 1-4, 1-20
- NEW subparameter (UR) 6-15
- NLUBDTF parameter (IS) 2-48

- NLUBPGM
 - worksheet (IS) 1-11
- NLUBPGM parameter (IS) 2-48
- No automatic cataloging (UR) 1-20
- No options in effect (UR) 5-6
- NOGVIS configuration option (IS) 5-23
- NOLOG option (UR) 3-4
- Non-BIM-EPIC datasets
 - deleting (IS) 6-58
- Non-generation datasets See Work datasets
- NOREW subparameter (UR) 6-27
- NOREWIND
 - DD statements (UR) 6-35
- NOTFND (IS) 6-15

O

- O option (UR) 5-11
- OF2 parameter (UR) 3-37
- OFFLINE command (UR) 2-6, 3-31
- Offload status (UR) 1-16
- Offload system
 - consolidating versions (UR) 13-34
 - offloading datasets (UR) 13-22
 - onloading datasets (UR) 13-27
 - selecting offload datasets (UR) 13-17
- OFFLOAD.DATA.SET (UR) 13-23
- Offload/Backup (UR) 1-31
- Offloaded datasets listing (UR) A-20
- Offloading datasets (UR) 13-22
- OLD subparameter (UR) 6-15
- OMIT
 - TSIDOFD (UR) 13-20
- One use only (UR) 3-35
- Online access to job accounting database
 - TOJA (UR) 11-13
- Online catalog management See Online Manager
- ONLINE command (UR) 2-6, 3-32
- Online Flow panel (UR) 2-16
- Online help (UR) 2-4
- Online Manager
 - accessing security (UR) 2-8
 - BIM-FAQS/ASO installation (IS) 2-70
 - catalog management (UR) 3-3
 - CICS installation (IS) 2-72
 - exiting (UR) 2-4
 - help (UR) 2-4, 2-16
 - installation requirements (IS) 2-2
 - Primary Option Menu (UR) 2-4
 - scratch tapes (UR) 12-2
 - user profiles (UR) 2-12
 - using (UR) 2-3

- VTAM access (IS) 2-71
- Online Manager (UR) 1-7
- Online VTOC displays (UR) 13-11
- Onloaded datasets
 - listing (UR) A-21
- Onloading datasets (UR) 13-27
- ONLREWCD configuration options (IS) 5-23
- OPEN command
 - TSIDASS (UR) 8-2
- OPEN not issued (UR) 7-35
- OPEN parameters (UR) 8-3
- Open version (UR) 1-16
- Operating system requirements (IS) 2-5
- Operator console (IS) 6-29
- Operator console commands (UR) 12-5
- OPT block size (UR) 6-13
- OPTION control statement
 - special options (IS) 5-29
- OPTION control statement (IS) **5-4**
- OPTION parameter
 - ALLMSGs (UR) 6-27
 - CAT (UR) 6-27
 - CISIZE (UR) 6-27
 - NOREW (UR) 6-27
 - RESET (UR) 6-27
 - REW (UR) 6-27
 - RUN (UR) 6-28
 - SPACE (UR) 6-28
 - SYSnnn (UR) 6-29
 - TPL (UR) 6-29
- OPTION parameter (UR) 6-8, 6-26, 6-38
- OPTION SYSPARM (UR) 10-8
- Options, special (IS) 5-29
- OPTI-TAPE ACL
 - configuration options (IS) 5-73
- OPTI-TAPE ACL support (IS) 5-73
- OR condition (UR) 9-9
- OS concatenation *See* Concatenating datasets
- Output
 - prime dataset (UR) 2-41
 - subdataset (UR) 2-41
- Output DTF names
 - TSIDDMP (UR) 8-16
 - TSIDDDTD (UR) 8-9
- Overriding
 - configuration options (IS) 6-27
 - cycles retention (UR) 3-20
 - dataset defaults (UR) 6-9
 - extent statements (UR) 7-8
 - revolving tape control (IS) 5-34
 - tape density (UR) 1-12
- Overriding normal cycles retention (UR) 1-25

- Overriding the DSN Catalog
 - block size (UR) 7-2
 - extent size (UR) 7-2
 - retention criteria (UR) 7-2
 - tape density (UR) 7-2
- Owner ID
 - single-digit (IS) 5-34
- Owner ID (UR) 1-12, 12-3
- Ownership (UR) 12-3

P

- P option (UR) 5-11
- PACCT (UR) 7-48
- PACKSCAN command (IS) 6-56
- Panel help (UR) 2-16
- Panel navigation help (UR) 2-16
- Parameter tables xii, xi
- Partition GETVIS (IS) 2-63
- Partition independence
 - disk pools (UR) 1-14, 1-31
- Partition independence (UR) **7-30**
- Partitions
 - deactivating BIM-EPIC in (IS) 5-18
 - deactivating/reactivating by (IS) 6-26
 - dynamic (IS) 5-18
 - starting/stopping (UR) 1-32
- PASS parameter (UR) 6-17
- PASSWD configuration option (IS) 5-23
- PASSWD configuration option (UR) 3-7
- PASSWORD command (UR) 3-7
- Passwords
 - EDD and version use (UR) 1-9
 - security (UR) 2-9
- PCT tables (IS) 2-76, 2-77
- PDISPLAY,POFFLOAD,Printer/Punch Tape Spooling
 - POWER (UR) 7-49
- Performance
 - master index (IS) 2-23
- PERM command (IS) 6-60
- Permanent retention (UR) 1-10
- Physical IOCS
 - inhibiting LUB allocation (UR) 5-7, 5-11
- Physical IOCS (UR) 1-17, 1-37, **7-35**
- Placing datasets on specific volumes (UR) 13-6
- PLTPI Table (IS) 2-90
- Pool ID specification
 - DD statements (UR) 6-35
- Pool names (UR) 1-14, 1-31, 6-29
- Pools *See* Disk pools and Tape pools
 - default (UR) 1-14
 - tape (IS) 2-39

POOL-x (IS) 5-67
 POOL-x (UR) 2-40
 Posted events See Event posting
 Post-installation procedures (IS) 2-91
 POWER
 queued jobs (UR) 10-4
 POWER (IS) 2-9
 POWER job accounting (UR) 7-48
 POWER PDISPLAY,POFFLOAD, Printer/Punch Tape
 Spooling (UR) 7-49
 Pre-4.1 Upgrade Checklist (IS) 1-4
 PREFIX configuration option (IS) 5-23
 Pre-Installation Worksheet (IS) 1-6
 Pre-open assignment checking
 TSIDSRT (UR) 8-6
 Pre-open assignment checking (UR) 1-37
 Pre-open checking (UR) 6-27
 Pre-open processing
 TSIDASS (UR) 8-2
 Prevent tape rewind
 DD statements (UR) 6-35
 Preventing BIM-EPIC from releasing the tape drive
 (UR) 5-7
 Preventing rewind with TLBL option 3 (UR) 5-7
 Primary extent size (UR) 1-14
 Primary Option Menu (UR) 2-4
 Prime datasets (UR) **1-12, 7-14**
 Print options
 TSIDMNT (UR) 3-4
 PRINT parameter (IS) 6-42
 Printer datasets
 DD statements (UR) 6-38
 Printer output (UR) 7-56
 Printing dataset labels (UR) 12-16
 Printing datasets
 TSIDDMPT (UR) 8-16
 Printing external volume labels (UR) 12-14
 Printing labels See Labels
 PRIVATE DD parameter (UR) 6-35
 Processing multiple datasets
 dataset drop (UR) 7-63
 Processing options See Configuration options
 Profiles
 \$DEFPROF (UR) 2-12
 \$DEFSEC (UR) 2-7
 \$EPIC (UR) 2-7, 2-12
 security (UR) 2-7
 user (UR) 2-12
 Program Audit panel (UR) 11-14, A-5
 Program names
 no LUB allocation (IS) 1-11
 reblocking with Undefined Record Format (IS) 1-11

Program Overview panel (UR) 11-13, A-4
 Programs
 inhibiting LUB allocation (IS) 2-47
 Projected Scratch (PSCRATCH) Report (UR) A-34
 PROJSER control statement (UR) 12-20
 PROJSLT control statement (UR) 12-20
 PRT control statement (IS) 2-35
 PSAM (IS) 5-31
 PSCRATCH (UR) 1-21
 Punch datasets
 DD statements (UR) 6-38
 PURBYP configuration option (IS) 5-23
 PURBYP configuration option (UR) 1-26
 PURGE command (UR) 2-5, 2-39, 2-43, 2-44, 3-33
 Purging
 EDD (UR) 3-33
 tape volumes (UR) 2-44
 versions (UR) 3-33
 volsers (UR) 3-33
 with DD statements (UR) 6-17
 Purging (UR) 1-35
 Purging datasets with JMCL (UR) 9-5

Q

QDAS
 access during open/EOX (UR) 1-42
 activating (IS) 5-52
 and space holder records (UR) 1-43
 deactivating (IS) 5-52
 space requirements (IS) 2-7
 TSIVDEF macro (IS) 2-55
 TSIVOFF (IS) 5-52
 TSIVON (IS) 5-52
 QDAS (IS) 2-52, 5-44
 Quick allocation with QDAS (UR) 1-42
 QUIESCE command (UR) 2-6, 3-34

R

R option (UR) 5-11
 Random access (UR) **4-5**, 4-8
 Random initialization
 example (UR) 12-12
 Random initialization (UR) 12-9
 RBLKPGM parameter (IS) 2-48
 Reading multiple datasets (UR) 7-61
 Reading VSE labels
 TSIDGLB and TSIDPLB (UR) 8-21
 Reblocking
 BLKSIZE DD parameter (UR) 6-14
 datasets processed by PIOCS (UR) 7-35
 DLBL option B (UR) 5-10

- EDD block size (UR) 1-10
- REC control statement (IS) 6-21
- Recatalog a deleted version (UR) 3-8
- RECMOVEn SAMPJOB (IS) 4-7
- RECORD configuration option (IS) 5-23
- RECORD configuration option (UR) 1-27, 11-1
- RECORD control statement (IS) 5-50
- Record length
 - LRECL DD parameter (UR) 6-14
- Recorder
 - activating (IS) 5-45
 - activation (IS) 2-94
 - deactivating (IS) 5-45
- Recorder File
 - BACKUP (IS) 5-46
 - comments (UR) 1-27
 - configuration options (UR) 1-27
 - creating (IS) 2-17
 - creation of job accounting database (IS) 2-85
 - daily maintenance JCL (IS) 2-91
 - formatting (IS) 2-23, 6-21
 - history dataset (UR) 11-2
 - job accounting reports (UR) 11-1
 - labels (IS) 2-58
 - location (IS) 2-21
 - maintaining (IS) 5-45
 - moving or changing size (IS) 4-7
 - multi-CPU environment (IS) 5-46
 - sharing (IS) 2-68
 - size requirement (IS) 2-21
- Recorder File (UR) 1-27
- Recorder subsystem (UR) 1-27
- RECOVERn SAMPJOB (IS) 4-8
- RECSHR configuration option (IS) 5-23
- RECSHR configuration option (UR) 1-27
- RECWARN configuration option (UR) 1-27
- RECWMMSG configuration option (UR) 1-27
- REF parameter (UR) 6-35
- Refer-back
 - multi-dataset processing (UR) 6-35
 - space allocation (UR) 6-27
 - subdatasets (UR) 6-35
- Regular maintenance (IS) 4-2
- Re-initializing tapes (UR) 1-30
- Relative version
 - DD statements (UR) 6-22
- RELEASE command (UR) 2-5, 2-27, 2-36, 3-35
- Release unused disk space
 - DD statements (UR) 6-30
- Releasing
 - EDD (UR) 2-27
 - GETVIS (UR) 4-9
 - jobs from POWER queue (UR) 10-4
- Releasing assignment
 - FREE statement (UR) 6-24
- Remote print, installation (IS) 2-88, 2-89, 2-90
- Removing
 - disk dataset from BIM-EPIC control (IS) 6-66
 - EDD (UR) 3-33
 - version information (UR) 2-29
 - versions (UR) 3-33
 - volsers (UR) 3-33
- RENAME command
 - TSIDVUT (IS) 6-62
- RENAME command (UR) 2-5, 3-36
- Renaming datasets (UR) 3-36
- RENUM command (IS) 6-64
- RENUMB command (IS) 6-66
- REPORT
 - TSIDOFD (UR) 13-20
- Report listing
 - creation date (UR) A-30, A-33
 - creation job (UR) A-30
 - creation job sequence (UR) A-33
 - display active (UR) A-30, A-33
 - display NEW (UR) A-33
 - display NEW report (UR) A-30
 - display PULL (UR) A-33
 - display PULL report (UR) A-30
 - display PULL, DSN report (UR) A-30
 - display scratch report (UR) A-33
 - display vault report (UR) A-30
 - expiration date (UR) A-30, A-33
- Report writers (UR) 7-35
- Reports
 - Activity Log Report (UR) A-6
 - Create Date/Time Activity (UR) A-24
 - Create Job/Time Activity (UR) A-13
 - creating (UR) 1-33
 - customized (UR) A-2
 - Dataset to Tape (UR) A-11
 - Dataset to Tape Report (UR) 12-24
 - offload tape datasets (UR) 13-27
 - online VTOC (UR) 13-11
 - Projected Scratch (PSCRATCH) (UR) A-34
 - Tape to Dataset (UR) A-23
 - Tape to Dataset Report (UR) 12-25
 - TSIDBRP (UR) A-8
 - TSIDDEB (IS) 6-25
 - TSIDDIS (UR) A-11
 - TSIDJAC (UR) A-13
 - TSIDLVT (UR) A-15
 - TSIDMNT (UR) A-17
 - TSIDOFD (UR) A-19

- TSIDOFLL (UR) A-20
- TSIDONL (UR) A-21
- TSIDPRV (UR) A-22
- TSIDSDS (UR) A-23
- TSIDTAC (UR) A-24
- TSIDUTL (IS) 6-7
- TSIDUTL (UR) 4-2
- TSIDVLT (UR) A-35
- vaulting cross-reference (UR) 12-18
- Vaulting Cross-Reference (UR) A-22
- Vaulting Cross-Reference Report (UR) 12-23
- VTOC (UR) 13-7
- REPRO (UR) 7-44
- Request handling (IS) 5-53
- Requirement for DTFDA datasets (UR) 7-10
- Requirements, operating system (IS) 2-5
- RERUN configuration option (IS) 5-23
- Resequencing dataset extents (UR) 13-5
- Resequencing index (UR) 4-5
- RESET parameter (UR) 9-13
- RESET subparameter (UR) 6-27
- Resizing the DSN Catalog See Maintenance procedures
- Resizing the Recorder File See Maintenance procedures
- RESTART subparameter (UR) 9-6
- Restarting from a checkpoint (UR) 7-54
- Restarting job
 - TYPRUN=RESTART (UR) 9-6
- RESTORE
 - IDCAMS (UR) 7-43
 - LIBR (UR) 7-46
- RESTORE command (IS) 6-13
- Restoring datasets (UR) 13-27
- Restricting BIM-EPIC control (UR) 1-5
- RETAIN
 - TSIDVUT (IS) 6-68
- Retain a Version panel (UR) 2-49
- RETAIN command
 - difference from CYCLE (UR) 2-50
- RETAIN command (UR) 2-5, 2-49, 3-37
- RETAIN parameter (UR) 6-35
- RETEN configuration option (IS) 5-24
- RETEN configuration option (UR) 1-20
- Retention
 - tape (UR) 12-2
- Retention
 - concepts (UR) 2-46
 - cycles (UR) 1-11
 - days after last access (DLA) (UR) 1-11
 - HDR1 labels (UR) 12-2
 - LABEL parameter (UR) 6-25
 - overriding (UR) 1-25
 - PERM (UR) 1-10
 - rules for Start Track 1 datasets (UR) 1-24
 - Start Track 1 datasets (UR) 1-4
 - TLBL statement (UR) 5-4
 - types of (UR) 1-10
- Retention criteria
 - modifying (UR) 1-25
 - SCRATCH (UR) 1-21
- Retention criteria (UR) 2-46
- Retention days
 - LABEL parameter (UR) 6-25
- Retention days (UR) 1-10
- Reusing same extent
 - MOD subparameter (UR) 6-15
- Reusing the same space (UR) 7-10
- REV parameter (UR) 12-3
- Revolving tape datasets (UR) 12-3
- Revolving tapes (UR) 1-12
- REW subparameter (UR) 6-27
- Rewind and unload with TLBL option 2 (UR) 5-7
- Rewind options
 - overriding (UR) 6-27, 6-28
- Rewind tape DD statements (UR) 6-35
- Rewind with TLBL option 1 (UR) 5-6
- RFBKUP (IS) 5-45
- RLSE parameter (UR) 6-30
- RMNTPRT (IS) 2-94
- RMNTPRT configuration option (IS) 5-24, 6-45
- RMNTPRT configuration option (UR) 12-16
- ROUND parameter (UR) 6-30
- ROUTE parameter (IS) 6-42
- RPG-II (UR) 7-56
- RSTRT (UR) 7-54
- RSTRT parameters (UR) 8-4
- Rules for dataset retention
 - multiple criteria (UR) 1-23
 - open tapes (UR) 1-23
 - work datasets (UR) 1-23
- Rules for dataset retention (UR) 1-22
- RUN subparameter (UR) 6-28
- Running
 - TVTO (UR) 13-11
- Running TSIDLVT (UR) A-15

S

- S option (IS) 5-33
- S option (UR) 5-12
- SAM datasets (UR) 1-5, 7-28
- Sample jobs (IS) 2-16
- SCAN subparameter (UR) 9-6

- Scanning JCL
 - TYPRUN parameter (UR) 6-3
- Scheduling dependent tasks (UR) 10-6
- Scheduling programs (IS) 6-50
- SCRATCH
 - TSIDUTL (UR) 1-21
- SCRATCH command (IS) 6-14
- SCRATCH command (UR) 2-5, 2-52, 3-39
- Scratch pools
 - synchronizing (IS) 5-67, 5-72, 6-7
- Scratch processing
 - work datasets (UR) 1-21
- Scratch processing (UR) **1-35**
- Scratch status (UR) 1-16, 2-41
- Scratch tapes (UR) 12-2
- Scratched tape versions (UR) 3-24
- Scratching
 - ACL cartridges (UR) 3-22
 - at close with TLBL option 4 (IS) 5-32
 - at close with TLBL option 4 (UR) 5-7
 - DD statements (UR) 6-17
 - old tape versions (UR) 1-10
 - open versions (UR) 1-38
 - tapes (UR) 3-22
 - versions (UR) 2-52
 - work datasets (UR) 1-22
- Scratching (UR) 2-46
- Scratching a version
 - DD statements (UR) 6-17
- Search Catalog panel (UR) 2-19, 2-30
- Searching the catalog (UR) 2-19
- Secondary allocations
 - at close (UR) 1-43
 - DD statement (UR) 6-35
 - DD statements (UR) 6-30
- Secondary allocations (UR) 13-3
- Secondary extent size (UR) 1-14
- Secondary extents
 - at close (UR) 1-43
 - fragmentation (UR) 13-2
- Secondary pooling. See Owner ID
- Secured Users panel (UR) 2-8
- Security
 - accessing (UR) 2-8
 - Online Manager (UR) 2-7
- Security profiles
 - \$DEFSEC (UR) 2-7
 - \$EPIC (UR) 2-7
 - creating (UR) 2-8
- Security profiles (UR) 2-7
- Security programs (IS) 6-48
- Selecting
 - offload datasets (UR) 13-17
 - SYS number (UR) 1-28
 - tapes (UR) 2-41
 - versions (UR) 2-32
- SEQ parameter
 - VAULT (UR) 12-21
- Sequential access
 - terminating (UR) 4-9
- Sequential access (UR) **4-5**, 4-9
- Sequential datasets
 - DD statements (UR) 6-14
- Sequential datasets (UR) 7-25
- Sequential initialization (UR) 12-9
- SER DD parameter (UR) 6-35
- Serial numbers (UR) 12-2
- Set condition code to zero See RESET
- SETC parameter (UR) 9-11
- SETG parameter (UR) 9-11
- SHARE configuration option (IS) 5-24
- SHARE configuration option (UR) 1-29
- Share files, QDAS (IS) 2-52, 2-68
- Sharing a DSN Catalog (UR) 2-2
- Sharing disk volumes (IS) 2-52
- Sharing resources (IS) 5-63, 5-68
- SHR subparameter (UR) 6-15
- Sign-On panel (UR) 2-10
- Silent mode (IS) 3-2
- Size See Tape size
- Skip JCL See Job management control language
- SLASIZE configuration option (IS) 5-24
- Slots
 - current (UR) 1-18
- SM2 See Sort packages
- SM-2 (UR) 7-37
- Small tapes (UR) 12-3
- Software compatibility (IS) 2-8
- SOL/VSE
 - Tape Output Functions (UR) 7-51
- SORT files (UR) 7-7
- Sort interface
 - TSIDSRT (UR) 8-6
- SORT packages
 - add dummy tape device (IS) 2-61
- Sort programs
 - CA-SORT (UR) 7-40
 - DFSORT (UR) 7-40
 - SYNCSORT (UR) 7-40
 - TSSRTDEF SORT macro (IS) 6-42
- Sort programs (UR) 7-35, 7-37
- SORTIN
 - non-standard (IS) 5-31

- SORTIN (IS) 5-31
- SORTOUT (IS) 5-31
- SORTWK datasets
 - LUB allocation (IS) 5-31
- SORTWRK datasets (UR) 7-31
- SPACE DD parameter (UR) 6-27
- Space holder records (UR) 1-43
- SPACE parameter
 - example (UR) 6-30
 - secondary extent size (UR) 13-3
- SPACE parameter (UR) 6-8, **6-30**
- Space requirements (IS) 2-7
- SPACE subparameter (UR) 6-28
- Special DD names (UR) 6-5
- Special options
 - descriptions (IS) 5-31
 - EPS022 (UR) 7-47
 - EPS043 (UR) 12-4
- Special options (IS) **5-29**
- Special terms xi, x
- Specifying
 - DD statements (UR) 6-2
 - JMCL parameters (UR) 9-2
 - tape size (UR) 12-3
- Standard Label Area (IS) 2-37
- Standard labels (IS) 2-58
- Standards
 - JCL (UR) 7-5
- START control statement (IS) 5-51
- START control statement (UR) 13-29
- Start Track 1 datasets
 - 7-day default retention (UR) 1-4
 - backing up (IS) 5-31
 - configuration options (UR) 1-4
 - converting to controlled (UR) 13-5
 - DD statements (UR) 6-9
 - DSKRET retention (IS) 5-38
 - extent sequence number (UR) 7-25
 - TYPEFLE=WORK
 - (UR) 7-32
 - VOLUME parameter (UR) 6-27
- Start Track 1 datasets (UR) **1-4**, 1-24, 7-9, 8-15
- Starting
 - BIM-EPIC (IS) 5-2
 - IBM ACL support (IS) 5-60
 - Memorex ATL support (IS) 5-63
 - STK ACL support (IS) 5-68
- Starting and stopping partitions (UR) 1-32
- Start-up options See Configuration options
- Status
 - device down (UR) 3-31
- STATUS command (UR) 12-6
- STEPCHAT statements
 - overriding (UR) 6-27
- STEPCHAT statements (UR) 6-5
- STEPLIB statements
 - unsupported (UR) 6-5
- STK ACL
 - configuration options (IS) 5-68
 - HSC definitions (IS) 5-72
 - requirements (IS) 2-6
- STK ACL support
 - activating and deactivating (IS) 5-68
- STOP
 - TSIDINIT (UR) 12-10
- Stopping
 - IBM ACL support (IS) 5-60
 - Memorex ATL support (IS) 5-63
 - STK ACL support (IS) 5-68
- STRTRK configuration option (IS) 5-24
- STRTRK configuration option (UR) 1-4, 13-5
- Subdataset status (UR) 1-16
- Subdatasets
 - autocatalog with DD statements (UR) 6-17
 - FREE DD statement (UR) 6-24
 - limitations (UR) 7-15
 - refer-back with DD statements (UR) 6-35
 - retention criteria (UR) 7-15
- Subdatasets (UR) 1-12, **7-15**
- Subtasks
 - initiating (IS) 5-53
- Summary display (UR) A-17
- Summary report
 - TSIDMNT (UR) A-17
- Summary Report
 - BIM-EPIC upgrade (IS) 2-3
- Summary Report (IS) 6-30
- SVA
 - size (IS) 2-62, 2-63
 - space requirements (IS) 2-7
- Symbolic replacements (UR) 9-12
- Synchronizing scratch pools (IS) 5-67, 5-72, 6-7
- SYNCIBM command (IS) 6-17
- SYNCMEM command (IS) 6-18
- SYNCOPT command (IS) 6-19
- SYNCSORT (IS) 2-10
- SYNCSORT (UR) 7-37
- SYNCSTK command (IS) 6-20
- SYS number (UR) 1-10
- SYS numbers
 - assigned by BIM-EPIC (UR) 1-28
- SYS000 (UR) 7-7
- SYS00n (UR) 7-33
- SYSABEND statements (UR) 6-5, 6-6

SYSCHK statements
 unsupported (UR) 6-5
 SYSCKEOV statements
 unsupported (UR) 6-5
 SYSID (IS) 6-15
 SYSIN statements (UR) 6-5
 SYSIPT (UR) 7-56, 7-59
 SYSIPT data
 JMCL parameter (UR) 9-5
 SYSLNK (UR) 7-33
 SYSLST (UR) 7-56, 7-59
 SYSMDUMP statements (UR) 6-5, 6-6
 SYSnnn DD parameter (UR) 6-27
 SYSnnn subparameter (UR) 6-29
 SYSOUT parameter (UR) 6-6, 6-38
 SYSOUT statements
 unsupported (UR) 6-5
 SYSPCH (UR) 7-56, 7-59
 SYSPRINT statements
 unsupported (UR) 6-5
 SYSSTORE macro (IS) 3-22, 3-28
 System datasets
 DD statements (UR) 6-26
 example (UR) 6-27
 open across job boundaries (UR) 5-11
 VSE device independence (UR) 7-59
 System GETVIS (IS) 2-63
 System ID
 searching by (UR) 2-19
 System ID (UR) 1-9
 System logical units
 ASSGN (UR) 8-2
 System Manager conversions *See* BIM-EPIC
 Conversion Guide
 System messages *See* Messages
 System options *See* Configuration options
 System performance
 master index (IS) 2-23
 SYSUDUMP statements (UR) 6-5, 6-6

T

T option (UR) 5-12
 TAID transaction (IS) 2-82
 TAPCUU configuration option (IS) 5-24
 Tape
 ownership (UR) 12-3
 positioning (UR) 7-12
 TAPE configuration option (IS) 5-24
 Tape configuration options (UR) 1-29
 Tape datasets
 grouping (UR) 1-9

 listing (UR) A-29
 names (UR) 1-9
 scratching (UR) 1-35
 updating from CMS (UR) 2-2
 Tape density (UR) 1-12
 Tape drives
 7-track (IS) 5-37
 adding in ASI procedure (IS) 2-61
 availability (UR) 1-29, 12-4
 Tape drives (UR) 1-29
 Tape error count (UR) 1-18
 Tape labels
 valid volume serial numbers (UR) 12-2
 Tape Output Functions
 SQL/VSE (UR) 7-51
 Tape pool group (IS) 2-45
 Tape pool ID
 searching by (UR) 2-19
 Tape pool ID (UR) 1-12
 Tape pools
 creating (IS) 2-39
 DD statements (UR) 6-29
 HSC definitions (IS) 5-72
 LMS definitions (IS) 5-67
 new datasets (UR) 6-29
 TSIDPOL syntax (IS) 2-45
 worksheet (IS) 1-10
 Tape pools (IS) **2-44**
 Tape pools (UR) 2-40, 12-3
 Tape Pools Worksheet (IS) 1-10
 Tape processing
 adding dummy tape device (IS) 2-61
 label printing (IS) 2-84
 labeled (UR) 7-11
 multi-dataset tapes (UR) 7-15
 unlabeled (UR) 7-11
 Tape size (UR) 1-13, 12-3
 Tape to Dataset Report (UR) 12-25, A-23
 Tape to disk (UR) 7-56
 Tape use count (UR) 1-18
 Tape vaulting. *See* Vaulting
 Tape Volume Information panel (UR) 2-42
 Tapes
 by current location/destination (UR) A-35
 cartridge (UR) 1-30
 cataloging (UR) 2-40
 close processing (UR) 1-41
 displaying information about (UR) 2-42
 end-of-volume processing (UR) 1-40
 error count (UR) 3-18
 initializing (UR) 1-30, 2-40, 12-9
 managing catalog information (UR) 2-43

- manually controlled vaulting (UR) 12-22
- moving to next vault (UR) 12-20, A-35
- open processing (UR) 1-40
- open status (IS) 5-31
- open status (UR) 1-22
- printing dataset labels (UR) 12-16
- purging (UR) 2-39, 2-44
- recording cleaning (UR) 2-45
- reporting vault locations (UR) 12-21
- required by a job (IS) 6-53
- revolving (UR) 12-3
- scratch (UR) 12-2
- scratching (UR) 3-22
- synchronizing scratch pools (IS) 6-7
- use count (UR) 3-18
- using Online Manager (UR) 2-40
- volume labels (UR) 1-30
- Tapes, output
 - controlled and uncontrolled (UR) 12-7
- TAPLUB configuration option
 - assigning SYS number (UR) 1-28
- TAPLUB configuration option (IS) 5-24
- TAPPOL configuration option (IS) 2-44, 5-25
- TCT Table (IS) 2-89
- Technical support xii, xi
- Terminating
 - sequential access (UR) 4-9
- Termination options
 - DISP DD parameter (UR) 6-17
- Test environment (IS) 2-4
- Test jobs (UR) 7-64
- TFVAULT See Vaulting
- THLDOPEN (IS) 6-15
- THLDOPEN (UR) 1-22, 1-38
- TIME (IS) 6-15
- TIME parameter (UR) 9-5
- TLBL option codes
 - 0 no options in effect (UR) 5-6
 - 1 or W rewind at close (UR) 5-6
 - 16 or D dataset drop (UR) 5-7, 7-63
 - 2 or U unload at close (UR) 5-7
 - 3 do not rewind (UR) 5-7
 - 32 conditional catalog (UR) 5-7, 7-64
 - 4 or R scratch work tape (UR) 5-7
 - 48 concatenation (UR) 5-7
 - 64 no LUB allocation (UR) 5-7
 - 8 or H hold assignment (UR) 1-30, 5-7
- TLBL option codes (UR) **5-6**
- TLBL statements
 - BIM-EPIC differences (UR) 5-2
 - default version (UR) 7-4
- TLBL statements (UR) **5-2**
- TOJA
 - navigating in (UR) 11-15
 - report examples (UR) A-3
- TOJA transaction
 - installation (IS) 2-84
- TOJA transaction (IS) 2-79
- TOJA transaction (UR) 11-10, **11-13**
- Total capacity (UR) 3-11
- TPAMODE configuration option (IS) 5-25
- TPL subparameter (UR) 6-29
- TPRE transaction (UR) 12-16
- TPRT transaction
 - activating (IS) 2-84
 - activating (UR) 12-16
 - DCT entry (IS) 2-88
 - deactivating (UR) 12-16
 - installation (IS) 2-84
 - PLT entry (IS) 2-90
- TPRT transaction (IS) 2-80, 6-47
- TRANSFER command (UR) 2-5, 3-41
- TRK parameter (UR) 6-30
- Truncation
 - inhibiting with DLBL option N (UR) 13-4
 - requesting with DD statements (UR) 13-4
 - RLSE parameter (UR) 6-30
- TSICMNT See also Catalog management
- TSIDAC2
 - calling procedure (UR) 4-6
 - search arguments and calls (UR) 4-8
- TSIDAC2 (UR) **4-5**
- TSIDAR (UR) 12-5
- TSIDARP
 - report (UR) A-6
- TSIDARP (UR) **11-3**
- TSIDASS
 - DITTO (UR) 7-47
 - dummy device in ASI procedure (IS) 2-61
 - inhibiting LUB allocation (UR) 5-7, 5-11
 - RSTRT (UR) 8-4
 - use device independence (UR) 7-59
- TSIDASS (UR) 7-35, **8-2**
- TSIDBLD (IS) **6-23**
- TSIDBPR
 - deactivating (UR) 12-16
- TSIDBPR (IS) 6-46
- TSIDBPR (UR) **12-16**
- TSIDBRP
 - reports (UR) A-8
- TSIDBRP (UR) **11-6**
- TSIDCCHK (UR) **10-6**, 10-11
- TSIDCDK
 - BCAT (IS) 2-23, 6-21

- CAT (IS) 6-21
- examples (IS) 2-23
- REC (IS) 6-21
- TSIDCDK (IS) 4-6, 4-7, **6-21**
- TSIDCHK (UR) **10-6**, 10-8, **10-9**
- TSIDCLN (IS) 2-92
- TSIDCLOC (UR) **10-5**, 10-11
- TSIDCMRG (IS) **6-70**
- TSIDCREL (UR) **10-4**, 10-11
- TSIDCUNL (UR) **10-7**, 10-11
- TSIDDEB
 - BIM-EPIC upgrade (IS) 2-3
 - deactivating/reactivating BIM-EPIC (IS) 6-26
 - exceptions (IS) 6-27
 - reporting capabilities (IS) 6-25
 - Summary Report (IS) 6-30
- TSIDDEB (IS) 5-2, 5-44, **6-24**
- TSIDDEL (UR) **8-7**
- TSIDDIS (UR) **12-24**, A-11
- TSIDDMP (UR) 7-47, **8-16**
- TSIDDTD (UR) 7-47, **8-9**, 11-2
- TSIDFAV (UR) **10-15**
- TSIDFAV macro (IS) **6-44**
- TSIDGLB (UR) **8-21**
- TSIDINIT See Tape initialization
 - VOLSER statements (UR) 12-11
- TSIDINIT (UR) 1-30, **12-9**
- TSIDJAC
 - report (UR) A-13
- TSIDJAC (UR) **11-8**
- TSIDJAC report examples (UR) A-13
- TSIDJLD (IS) 2-85
- TSIDJLD (UR) **11-10**, 11-13
- TSIDLBL
 - example (UR) 12-14
- TSIDLBL (UR) 1-30, **12-14**
- TSIDLOC (UR) **10-5**, 10-8, **10-9**
- TSIDLST (IS) **6-53**
- TSIDLVT
 - control statements (UR) 13-8
 - report (UR) A-15
 - VTOC data in batch (UR) A-15
- TSIDLVT (UR) 1-31, **13-8**
- TSIDMAP
 - phase creation (IS) 2-39
 - worksheet (IS) 1-11
- TSIDMAP macro
 - END parameter (IS) 2-48
 - example (IS) 2-50
 - ILIST parameter (IS) 2-48
 - ILIST parameter (UR) 1-5
 - NLUBDTF parameter (IS) 2-48
 - NLUBPGM parameter (IS) 2-48
 - RBLKPGM parameter (IS) 2-48
 - syntax (IS) 2-47
- TSIDMAP macro (IS) 2-47
- TSIDMNT See also Catalog management
 - JCL requirements (UR) 3-4
 - OFFLINE (UR) 3-31
 - ONLINE (UR) 3-32
 - print options (UR) 3-4
 - QUIESCE (UR) 3-34
 - reports (UR) A-17
- TSIDOFB
 - examples (UR) 13-36, 13-37
- TSIDOFB (UR) **13-34**
- TSIDOFD
 - CATALOG (UR) 13-20
 - report (UR) A-19
 - REPORT (UR) 13-20
 - report examples (UR) A-19
- TSIDOFD (UR) **13-17**
- TSIDOFF (IS) **5-44**
- TSIDOFFL
 - examples (UR) 13-25
 - report (UR) A-20
 - report examples (UR) A-20
- TSIDOFFL (UR) **13-22**
- TSIDOJA.C (UR) 11-10
- TSIDON
 - example (IS) 2-94
 - OPTION (IS) 5-4
- TSIDON (IS) **5-2**
- TSIDONL
 - examples (UR) 13-30
 - report (UR) A-21
 - report examples (UR) A-21
- TSIDONL (UR) **13-27**
- TSIDPCS (IS) 6-50
- TSIDPEN (UR) 12-16
- TSIDPLB (UR) **8-21**
- TSIDPOL
 - designing pools (IS) 2-40
 - examples (IS) 2-43
 - phase creation (IS) 2-39
 - statement creation (IS) 2-40
 - syntax (IS) 2-42
 - tape pool syntax (IS) 2-45
- TSIDPOL phase
 - TSIDMAP macro (IS) 2-47
- TSIDPR (IS) 6-46
- TSIDPRT (IS) 6-47
- TSIDPRT (UR) **12-16**
- TSIDPRV

- report (UR) A-22
- report examples (UR) A-22
- TSIDPRV (UR) 12-18, **12-23**
- TSIDREC.C (UR) A-2
- TSIDREL (UR) **10-4**, 10-8, **10-9**
- TSIDRFS
 - BACKUP (IS) 5-46
 - CLEAR (IS) 5-47
 - comments (UR) 1-27
 - END (IS) 5-48
 - MAINT (IS) 4-2, 5-49
 - moving/changing Recorder File (IS) 4-7
 - RECORD (IS) 5-50
 - required maintenance (IS) 4-2
 - START (IS) 5-51
- TSIDRFS (IS) **5-45**
- TSIDSDS
 - report example (UR) A-23
- TSIDSDS (UR) **12-25**
- TSIDSEC (IS) 6-48
- TSIDSMNT
 - CTL (IS) 2-32
 - DEL (IS) 2-33
 - FMT (IS) 2-34
 - JCL examples (IS) 2-36
 - PRT (IS) 2-35
- TSIDSMNT (IS) **2-26**
- TSIDSMNT program
 - ACLOFFL (IS) 2-27
 - ACLONL (IS) 2-28
 - ADD (IS) 2-29
- TSIDSRT
 - non-standard SORTIN (IS) 5-31
- TSIDSRT (UR) 7-35, **8-6**
- TSIDTAC
 - report (UR) A-24
- TSIDTAC (UR) **11-9**
- TSIDUNL (UR) **10-7**, 10-8, **10-9**
- TSIDUTL
 - ACLEJECT (IS) 6-5
 - BACKUP (IS) 6-6
 - BDSPLY (IS) 6-7
 - dataset requirements (IS) 6-2
 - DSPLY (IS) 6-7
 - DSPLY (UR) 4-2, 12-21, 12-23
 - DSPLY SCRATCH (UR) 12-2
 - DSPLY. Also see Reports
 - FASTBU (IS) 6-10
 - FASTRS (IS) 6-11
 - INDEX (IS) 6-12
 - INDEX (UR) 4-5
 - projected scratch tapes (IS) 6-14
 - PSCRATCH (UR) 1-21, 12-2
 - reporting locations of vaulted tapes (UR) 12-23
 - RESTORE (IS) 6-13
 - SCRATCH (IS) 4-2, 6-14
 - SCRATCH (UR) 1-10, 1-21, 12-2
 - SYNCIBM (IS) 6-17
 - SYNCMEM (IS) 6-18
 - SYNCOPT (IS) 6-19
 - SYNCSTK (IS) 6-20
 - VAULT (UR) 12-23
 - VLT parameter (UR) 12-23
- TSIDUTL (IS) 4-2, **6-2**
- TSIDUXI (IS) 6-51
- TSIDVLT
 - excludes from Recorder (UR) 1-27
 - reports (UR) A-35
- TSIDVLT (UR) 12-17, **12-20**
- TSIDVUT
 - DELETE (IS) 6-58
 - PACKSCAN (IS) 6-56
 - PERM (IS) 6-60
 - RENAME (IS) 6-62
 - RENUM (IS) 6-64
 - RENUM (UR) 7-47
 - RENUMB (IS) 6-66
 - resequencing extents (UR) 13-5
 - RETAIN command (IS) 6-68
- TSIDVUT (IS) **6-54**
- TSIDVUT (UR) 1-25, 1-31
- TSIVCLR, creating QDAS share files (IS) 2-53
- TSIVDEF
 - defining (IS) 2-55
 - example (IS) 2-56
- TSIVOFF (IS) **5-52**
- TSIVON (IS) **5-52**
- TSPR installation (IS) 2-89
- TSSRTDEF sort macro (IS) 6-42
- TSTJOB configuration option (IS) 5-26
- TSTJOB configuration option (UR) 7-64
- Turning off
 - BIM-EPIC (IS) 5-44
 - IBM ACL support (IS) 5-60
 - Memorex ATL support (IS) 5-63
 - QDAS (IS) 5-52
 - Recorder (IS) 5-45
 - STK ACL support (IS) 5-68
- Turning on
 - BIM-EPIC (IS) 5-2
 - IBM ACL support (IS) 5-60
 - Memorex ATL support (IS) 5-63
 - QDAS (IS) 5-52
 - Recorder (IS) 5-45

STK ACL support (IS) 5-68
 TVTO (UR) 1-31, **13-11**
 TVTO installation (IS) 2-84
 TVTO transaction (IS) 2-81
 TYPEFLE=WORK
 control on disk only (UR) 1-5
 DLBL option N required (UR) 5-11
 do not use DLBL option F (UR) 5-10
 secondary allocation unavailable (UR) 13-3
 TYPEFLE=WORK (IS) 5-33
 TYPEFLE=WORK (UR) **7-32**
 TYPRUN parameter
 SCAN (UR) 6-3
 TYPRUN parameter (UR) 9-6

U

UIN (IS) 5-31
 UIN (UR) 7-7
 UNCDSK configuration option
 ISAM limitation (UR) 7-26
 uncontrolled DA datasets (UR) 7-25
 UNCDSK configuration option (IS) 5-26
 UNCDSK configuration option (UR) 1-5, 6-30, 7-34
 Uncontrolled datasets
 DD statement example (UR) 6-9
 DD statements (UR) 6-9, 6-30
 DTF names (IS) 1-11
 SAM (UR) 7-28
 specifying (IS) 2-47
 using DD statements (UR) 6-30
 VSAM (UR) 7-28
 Uncontrolled datasets (UR) 2-41, 7-34
 Uncontrolled tapes
 Early Drive Release (UR) 1-29
 UNCTAP configuration option
 Early Drive Release (UR) 1-29
 UNCTAP configuration option (IS) 5-26
 UNCTAP configuration option (UR) 1-5, 7-34, 12-7
 UNCTL configuration option (IS) 5-26
 UNCTL configuration option (UR) 1-5, 7-34
 Unit Control Blocks (IS) 2-25
 UNIT DD parameter (UR) 6-7
 UNIT parameter (UR) 6-8, 6-33, 6-38
 Unit record
 DD statements (UR) 6-26
 DTF (UR) 6-2
 UNLABEL configuration option (IS) 5-26
 Unlabeled status (UR) 1-16
 UNLOAD configuration option (IS) 5-26
 Unload tape
 DD statements (UR) 6-35

UNLOCK command (UR) 2-5, 2-28, 2-37, 3-42
 Unlocking
 datasets or versions (UR) 3-35
 EDD (UR) 2-28
 Unsupported ddnames (UR) 6-5
 Unsupported processing
 DITTO (UR) 7-2
 IJSYSNR (UR) 7-2
 IJSYSPC (UR) 7-2
 POWER LST (UR) 7-2
 POWER offload (UR) 7-2
 xxJSYSRS (UR) 7-2
 UOUT (IS) 5-31
 UOUT (UR) 7-7
 Update an EDD panel (UR) 2-25
 UPDATE command
 VLT parameter (UR) 12-19
 UPDATE command (UR) 2-5, 2-25, 2-48, 3-43
 Updating
 confirmation options (UR) 2-15
 EDD display format (UR) 2-13
 EDD information (UR) 2-25
 version display format (UR) 2-14
 VTOC entries (IS) 6-54, 6-56
 VTOC entries (UR) 1-31
 Updating SVA size (IS) 2-62, 2-63
 Upgrading
 DSN Catalog (IS) 2-92
 over more than one release (IS) 2-3
 preparation (IS) 2-3
 UPSI values (IS) 6-2
 Use device independence (UR) 7-59
 User exits
 supporting multiple job exits (IS) 2-65
 User exits (IS) 6-48
 User ID
 searching by (UR) 2-19
 User ID (UR) 1-9
 User profiles
 \$DEFPROF (UR) 2-12
 \$EPIC (UR) 2-12
 confirmation options (UR) 2-15
 EDD display format (UR) 2-13
 version display format (UR) 2-14
 User profiles (UR) **2-12**
 User Profiles panel (UR) 2-12
 User Security panel (UR) 2-9
 USERID (IS) 6-15

V

V option (UR) 5-12

- VATTACH (IS) 3-2
- VATTACH (UR) 1-29
- VATTACH, VDETACH See Autoattach
- Vault locations (UR) 12-17
- Vault method number (UR) 1-13
- Vault Method Report (UR) 12-23
- VAULT parameter (UR) 12-21
- VAULT sequence (UR) 12-21
- VAULTDEF macro Also see Vaulting
 - assembling and linking (IS) 6-41
 - AVAULTS (UR) 12-22
- VAULTDEF macro (IS) **6-39**
- VAULTDEF macro (UR) 12-17
- Vaulting
 - assembling and linking macros (IS) 6-41
 - assigning vaulting methods (UR) 12-19
 - creating methods (IS) 6-40
 - creating vaults (IS) 6-39
 - cross-reference report (UR) 12-18
 - Cross-Reference Report (UR) 12-23
 - METHOD macro (IS) 6-40
 - METHOD macro (UR) 12-17
 - MOVE (UR) 3-30
 - MOVESER (UR) 12-20
 - MOVESLT (UR) 12-20
 - PROJSER (UR) 12-20
 - PROJSLT (UR) 12-20
 - reporting tape location (UR) 12-21
 - reports (UR) A-35
 - TSIDVLT (UR) 12-20
 - VAULTDEF macro (IS) 6-39
 - VAULTDEF macro (UR) 12-17
- Vaulting (IS) **6-38**
- Vaulting Cross-Reference Report (UR) 12-23, A-22
- Vaulting method assignment (UR) A-22
- Vaulting methods (UR) 12-17
- Vaults
 - current (UR) 1-18
 - manually controlled (UR) 12-22
- VDETACH (IS) 3-2
- VDETACH (UR) 1-29
- Version attributes
 - block count (UR) 1-17
 - block size (UR) 1-17
 - creating job and device (UR) 1-18
 - creation date and time (UR) 1-18
 - current vault and slot (UR) 1-18
 - expiration date (UR) 1-18
 - extent number and address (UR) 1-17
 - file sequence number (UR) 1-17
 - generation number (UR) 1-16
 - last access (UR) 1-18
 - record format (UR) 1-17
 - record length (UR) 1-17
 - reel number (UR) 1-17
 - status (UR) 1-16
 - tape error count (UR) 1-18
 - tape use count (UR) 1-18
 - version number (UR) 1-16
 - volume serial number (UR) 1-17
- Version Information panel (UR) 2-33
- Version specification
 - DD statements (UR) 6-22
- Version status
 - active (UR) 1-16
 - conditional catalog (UR) 1-16
 - lock (UR) 1-16
 - MCAT (UR) 1-16
 - offload (UR) 1-16
 - open (UR) 1-16
 - scratch (UR) 1-16
 - subdataset (UR) 1-16
 - unlabeled (UR) 1-16
 - work (UR) 1-16
- Version-level data (UR) 1-16, A-26
- Versions
 - adding (UR) 2-34, 3-8
 - changing dataset name (UR) 3-41
 - closing (UR) 2-38
 - CYCLE (UR) 2-50
 - default (UR) 7-4
 - deleting (UR) 3-22
 - display format (UR) 2-14
 - displaying (UR) 2-30, 2-33
 - locking (UR) 2-35
 - moving (UR) 3-41
 - online management (UR) 2-30
 - releasing (UR) 2-36
 - removing (UR) 2-39, 3-33
 - removing from catalog (UR) 2-29
 - resequencing (UR) 2-46
 - retention (UR) 2-49
 - scratching (UR) 2-52
 - selecting from a list (UR) 2-32
 - unlocking (UR) 2-37
- VGSFTAPE configuration option (IS) 5-27
- Viewing
 - job accounting database (UR) A-3
- VLT parameter (UR) 12-21
- VM
 - IUCV facility (IS) 5-63, 5-68
 - modifications See Autoattach
 - SVA requirement (IS) 2-62
 - VSE guests (IS) 5-65, 5-70

VM configuration option
 CP commands (UR) 9-20
 VM configuration option (IS) 3-2, 5-27
 VM Directory
 HSC server entry (IS) 5-69
 LMS server entry (IS) 5-64
 VOL DD parameter (UR) 6-7
 VOL parameter (UR) 6-35
 VOL1 labels (UR) 12-2
 Volser
 TSIDLBL (UR) 12-14
 VOLSER statements
 TSIDINIT (UR) 12-11
 VOLUME DD parameter (UR) 6-35
 VOLUME parameter
 number of secondary extents (UR) 13-3
 Start Track 1 datasets (UR) 6-27
 VOLUME parameter (UR) 6-8
 Volume serial number
 vault sequences (UR) A-29
 Volume serial numbers
 000000 (UR) 2-40
 printing labels (UR) 12-14
 ranges (UR) 2-40
 unique (UR) 2-40
 Volume serial numbers (UR) 12-2
 Volume serial request DD statements (UR) 6-35
 Volumes
 removing
 (UR) 3-33
 VSAM configuration option (IS) 5-28
 VSAM configuration option (UR) 1-5, 7-28
 VSAM datasets
 AMORG DD parameter (UR) 6-12
 AMP DD parameter (UR) 6-12
 BUFSP DD parameter (UR) 6-12
 DD statements (UR) 6-10, 6-26
 extent information (UR) 1-5
 not eligible for dataset drop (UR) 6-3
 VSAM datasets (UR) 7-28
 VSAM/SAM datasets
 DD statements (UR) 6-26
 VSE
 Attention Routine (UR) 12-5
 DASDSHR parameter (IS) 2-5
 guest machine (IS) 5-60, 5-63, 5-65, 5-68, 5-70
 JA parameter (IS) 2-5
 LOGSRC option (UR) 5-4
 Operator console commands (UR) 12-5
 system utilities (UR) 7-35
 VSE label (IS) 6-51
 VSE label area

 accessing (UR) 8-21
 TSIDGLB (UR) 8-21
 TSIDPLB (UR) 8-21
 VSE Label Area (UR) 1-40
 VSE/ESA 1.3
 space requirements (IS) 2-7
 VSE/ESA 2.x
 space requirements (IS) 2-7
 VSE/ICCF Interactive Partition GETVIS (UR) 7-52
 VTAM
 accessing Online Manager (IS) 2-71
 VTOC
 access during open/EOX (UR) 1-42
 generation datasets (UR) 1-16
 space holder records (UR) 1-43
 Start Track 1 datasets (UR) 1-4
 updates during close (UR) 1-43
 VTOC (UR) 1-42
 VTOC entries
 deleting (IS) 6-58
 display and update (IS) 6-56
 displaying (UR) 1-31, A-15
 updating (IS) 6-54
 VTOC online report, installing (IS) 2-84
 VTOC utilities
 batch reports (UR) 13-7
 TVTO (UR) 13-11
 VTOC utilities (UR) 1-31

W

Warnings

TIME overridden by SETIME (UR) 9-5

WARNINGS

ACCEPT=YES lets operator select tapes (UR) 12-8

ADD does not recover a VTOC entry (UR) 3-8

BIM-GSS must be installed before BIM-EPIC (IS)
2-2

Do not delete encrypted versions (UR) 1-9, 3-22

Do not issue VSE DVCUP for dummy device (IS)
2-61

Do not purge encrypted versions (UR) 3-33

Do not reblock a backup dataset (IS) 6-6

Do not reblock backup dataset (IS) 6-10

Do not rename prime or offload datasets (UR) 3-36

Do not use PURGE for routine deletion (UR) 3-33

EPS011=YES significantly increases BIM-EPIC
overhead (IS) 5-32

Install BIM-GSS before BIM-EPIC (IS) 1-6

Keep Recorder active (UR) 1-27

MAP values when SHARE=YES (IS) 2-55

MOVEs not shown on TSIDVLT reports (UR) 3-30

MOVEs nullified by auto vaulting (UR) 3-30
 MULTCPU=YES to share catalog (IS) 5-22
 Must have QDAS Share File on shared volume (IS)
 2-53, 5-55, 6-19
 No concatenation code on last label statement (UR)
 7-61
 No other job or CPU during FASTRS (IS) 6-11
 QDAS must be active for all sharing machines (IS)
 5-52
 RENAME won't check for duplicate names (IS) 6-
 62
 Reset Recorder whenever catalog backed up (IS) 4-
 2
 SHARE=YES to share among physical CPUs (IS)
 5-24
 Tape reuse before scratch (UR) 1-22
 TSIDPLB doesn't validate label data (UR) 8-22
 TSIDVUT for updating uncataloged only (IS) 6-54
 UPSI x1 restores oldest as version 1 (UR) 13-23
 Use different devices for DSN Catalog and Recorder
 File (IS) 2-20
 Use RECORD=YES (IS) 5-23
 Without version specified, DSN concatenates (UR)
 6-22
 Work dataset status (UR) 1-16
 Work datasets

autocatalog with DD statements (UR) 6-17
 DD statements (UR) 6-22
 scratching (UR) 1-22
 temporary datasets (UR) 7-29
 Work datasets (UR) **1-10**, 1-21
 Work files
 deleting (UR) 8-7
 Worksheets Also see Checklists. *Also see* Checklists
 configuration options (IS) 1-12
 Initial Installation Checklist (IS) 1-2
 no LUB allocation (IS) 1-11
 Pre-4.1 Upgrade Checklist (IS) 1-4
 Pre-Installation (IS) 1-6
 tape pools (IS) 1-10
 TSIDMAP (IS) 1-11
 uncontrolled DTF names (IS) 1-11

X

X option (UR) 5-12
 xJSYSRS (UR) 1-5

Z

Zeroing use and error counts (UR) 3-18